Copyright Information

This is a post-peer-review, pre-copyedit version of the following paper

You are welcome to cite this work using the following bibliographic information:

Cooperation between Autonomous Underwater Vehicle Manipulations Systems with Minimal Information Exchange

Enrico Simetti, Giuseppe Casalino, Ninad Manerikar, Alessandro Sperindé, Sandro Torelli and Francesco Wanderlingh
Interuniversity Center on Integrated Systems for the Marine Environment
Department of Informatics, Bioengineering, Robotics and Systems Engineering
University of Genova
Via Opera Pia 13, 16145 Genova, Italy
Email: enrico.simetti@unige.it

Abstract—The paper describes a novel cooperative control policy for the transportation of large objects in underwater environments using two UVMS (Underwater Vehicle Manipulator Systems). Due to the low bandwidth available in underwater scenarios, the main feature of the paper lies in the fact that the cooperative transportation of the commonly grasped object is carried out successfully by just exchanging the tool frame velocities at each sampling instant. A disturbance compensation technique is also presented to cope with sea currents and vehicle velocity tracking errors.

I. Introduction

In the last two decades a lot of work has been carried out in the field of underwater robotics related to subsea exploration and observation, mining and manipulation. There have been tremendous advancements in the field of underwater manipulation that can be seen from the completion of several projects including AMADEUS [1], ALIVE [2], SAUVIM [3] which successfully demonstrated autonomous manipulation capabilities. Then, within the recently concluded EU-funded project TRIDENT [4], more enhanced control capabilities for autonomous floating vehicle-arm systems were finally achieved for vehicles and arms of comparable masses and inertias and within a unifying control and coordination framework. By taking into account these developments, the notion of having two UVMS (underwater vehicle manipulator systems) performing autonomous intervention activities in a cooperative manner certainly comes to mind. To this regard, the national Italian project MARIS [5] has been launched recently and its main aim is to develop state of the art cooperative control algorithms as well as an experimental proof of concept trial, paving the way for future developments that would allow cooperative intervention to be used even in underwater environments.

Within this context, the contribution of this paper is the development, within a unified framework, of a cooperative control policy relevant to all the involved mission phases i.e grasping, transportation, and the final positioning of the shared
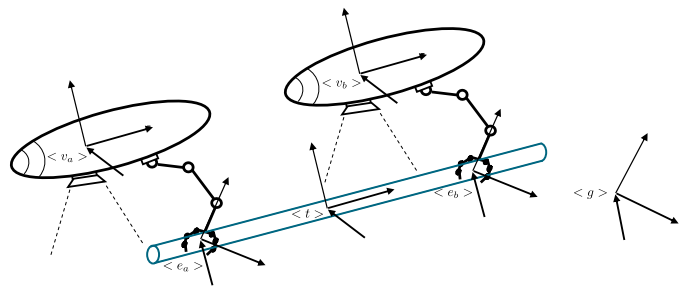

Fig. 1. The two cooperative UVMS and their relevant frames

object grasped by two systems. Figure 1 shows the two UVMS and their relevant frames.

The paper is structured as follows. Section II recalls the control objectives that a single floating manipulator has to tackle and that have been solved in [4]. Section III extends the approach to the mentioned decentralized multi objective constrained cooperative control problem. The simulation results supporting the proposed approach are presented in Section IV. Finally, some conclusions and foreseen future works are discussed in Section V.

II. Task Priority Based Coordinated Control for Individual Floating Manipulation System

A single floating manipulation system, during the execution of its assigned mission, has to comply with different safety and operability constraints. Given a reference mission where the UVMS has first to grasp an object, then transport it to a goal location and finally positioning it on the desired target location, the control objectives that it would need to tackle are listed in the following:

1) Joint Limits: the arm needs to maintain its joints within well defined bounds to avoid hitting their mechanical limits;
2) Manipulability: the arm must operate with a good dexterity, avoiding singular configurations;

3) Horizontal Attitude: the vehicle should avoid configurations with high tilt angles;
4) Camera centering for object transportation: this objective is activated during the transportation phase, and requires the goal frame grossly maintained within the visual cone of the endowed forward-looking camera;
5) Camera centering for object positioning: this task is activated during the final positioning phase and requires the grasped object frame and the goal one both maintained in the field of view of the endowing positioning camera;
6) Positioning camera distance and height: this objective is activated during the positioning phase and requires the positioning camera to be above a certain height and below a given horizontal distance from the goal frame; in order to enable the best operability conditions for the positioning camera;
7) Vehicle motion minimality: since the vehicle generally exhibits a larger mass and inertia than the arm, during the final positioning phase it is advisable to have it move for only the strict necessary amount needed for accomplishing the related tasks.

All the above objectives do not need to be precisely achieved. They are satisfied whenever the system is within certain configuration regions, for example whenever their relevant variable is below, above or between some given, pre-defined, thresholds. They are thus constraints of inequality type and all of them are related either to the safety of the system or aimed at guaranteeing its good operability.

Naturally, the system must also comply with objectives that must be satisfied exactly:

1) End effector object grasp: during the initial grasping phase, the UVMS must place the arm end-effector above the object to be grasped in a given grasping position;
2) Grasped object positioning: this objective is active during both the transportation and final positioning phase. It requires that the object is precisely positioned in a given position, i.e. that the tool-frame reaches the goal-frame of Fig. 1.

In order to execute many objectives concurrently a possible solution is the employment of the task priority framework [6], where each task is assigned a priority according to its importance. Hence the various tasks are solved accordingly to their priority in a descending order, with the highest priority task solved first and the solutions of the other tasks are solved inside the null space of the higher priority ones. An important limitation of the above mentioned framework is the inability of handling inequality constraints, which are converted to equality ones and cannot be properly activated and deactivated when needed, thus over-constraining the system.

An extension of the task-priority framework has been developed in [4], where suitable activation functions are used for the activation and deactivation of the inequality tasks. This means that when a task disappears from the priority list it results in an enlargement of the mobility space which in turn is favourable to the lower priority tasks.

The problem of cooperatively acting underwater floating manipulators can be solved by exploiting the task priority paradigm implemented in [4] and adapting it to the specific problems that the cooperation poses. The next section focuses on this extension.

III. EXTENSION TO COOPERATIVELY OPERATING UNDERWATER FLOATING MANIPULATORS

Once the two floating manipulators have firmly grasped a shared object, as indicated in Fig.1, the problem of transporting the object, and finally positioning its attached frame $< t >$ on the goal frame $< g >$, can be solved by using the same algorithmic structure as shown in [4]. However, in the cooperative scenario, two additional tasks are needed:

1) vehicles distance: maintaining a minimum distance between vehicles to avoid collisions;
2) object kinematic constraint: the two end-effectors velocities, when transferred to the common object frame, should be equal to avoid any stress on the object.

The cooperative control problem might be optimally solved by simply extending the application of the previously outlined task priority based individual control law to both UVMS in a centralized manner, as the presence of the object kinematic constraint inevitably couples the optimization problem of the two systems. However, this would require the exchange of all the Jacobians at every sample time in order to compute the centralized solution for both systems, which in the underwater case is not possible due to severe bandwidth limitations.

For this simple reason, the solution to the cooperative problem must be found minimizing the information exchange between the two UVMS. The extension to cooperatively operating underwater floating manipulators is thus presented in the following.

A. Cooperating UVMS with Minimal Information Exchange

The algorithm is divided in three separate steps: i) a first independent optimization, ii) the exchange of end-effector Cartesian velocities and iii) the re-computation of the system velocities.

1) First Independent Optimization: To minimize the information exchange, we first consider each UVMS as if it were the sole one acting on the object. A separate, independent optimization is performed, following the task priority algorithm [4] with the goal of satisfying the control objectives presented in Section II and the additional constraint of the minimum distance between the vehicles. We remark how in such a prioritized list of objectives, the end-effector position control has the lowest priority, and that the object kinematic constraint is not considered explicitly.

With this in mind, at each time instant, each system independently determines its own system control law, optimizing its own list of prioritized tasks as if it were the sole agent acting on the object. The two UVMSs obtain, in a fully decentralized way, the couple of system control actions $\dot{y}_a$ and $\dot{y}_b$, where each of the $\dot{y}$ is the stacked vector of the arm joint velocities $\dot{q}$ and vehicle velocities v.

The corresponding tool frame velocities for both the UVMS can be thus calculated as follows:

$$\dot{x}_a = J_a \dot{y}_a \tag{1}$$
$$\dot{x}_b = J_b \dot{y}_b \tag{2}$$

where J_a and J_b are the overall tool frame $< t >$ Jacobian matrices for systems a and b.

It must be noted how in general it might result in $\dot{x}_a \neq \dot{x}_b$, despite both UVMS having an end-effector velocity task with the same common velocity reference $\dot{\bar{x}}$. This is due to the fact that the end-effector task is the last one in the hierarchy and thus might not have the necessary DOFs available. This fact would result in the violation of the object kinematic constraint, with inevitable high stress on the object. To avoid this problem, the object kinematic constraint must be enforced, even if not in a explicit way. The next section focuses on this exact problem.

2) Exchange of Tool-Frame Velocities: For the above outlined reasons, the second step of the procedure is the exchange of the separately evaluated individually optimal tool-frame velocities $\dot{x}_a$, $\dot{x}_b$. Then, a common value $\dot{\hat{x}}$ is determined as the result of an a-priori agreed fusion policy, which may trivially be the mean value between $\dot{x}_a$, $\dot{x}_b$ or more generally a suitable convex combination of them. In a broader sense, it should be noted that such convex combinations could be used where one system can be the leader and the other system can be the follower and they can even smoothly change the role depending on the scenario.

In any case, the basic idea is to consider the individually optimal tool-frame velocities $\dot{x}_a$, $\dot{x}_b$ as somehow representative of the general state and needs of each of the two UVMS. For example, if both of them are equal to the desired tool frame velocity $\dot{\bar{x}}$, it means that both UVMS are able to execute all the tasks and the object kinematic constraint is already satisfied due to the common velocity reference. Conversely, when one of the tool-frame velocities differs from the desired one, then that UVMS clearly does not have enough DOFs to execute all its tasks and at the same time obtain the desired $\dot{\bar{x}}$. This consequently means that the object kinematic constraint is not automatically satisfied.

The computation of the common $\dot{\hat{x}}$ is a way to move the object with a velocity that allows both system to carry out their safety and operational-enabling tasks in a distributed way, avoiding the direct employment of the kinematic constraint.

3) Second Independent Optimization: After the exchange of the tool-frame velocities, both UVMS have agreed on a common $\dot{\hat{x}}$ and must now track this velocity as best as possible. For this reason, this second independent optimization is performed with the same tasks as the first one, where now the end-effector velocity tracking task is now at the top of the priority list, and is parametrized by the $\dot{\hat{x}}$ velocity reference in lieu of $\dot{\bar{x}}$ used in the first optimization run.

Being the first task of the hierarchy, this allows the full exploitation of all the DOFs of the system for its exact tracking and consequently means that the Jacobian matrix is always full row rank. Subsequent tasks can optimize the system velocity

without changing the obtained tool-frame velocity. Again, the two UVMSs obtain, in a fully decentralized way, the couple of system control actions:

$$\dot{\hat{y}}_a \triangleq \begin{bmatrix} \dot{\hat{q}}_a \\ \hat{v}_a \end{bmatrix}, \dot{\hat{y}}_b \triangleq \begin{bmatrix} \dot{\hat{q}}_b \\ \hat{v}_b \end{bmatrix} \tag{3}$$

However, the tool-frame velocities are this time

$$\dot{x}_a = J_a \dot{\hat{y}}_a = \dot{\hat{x}} \tag{4}$$
$$\dot{x}_b = J_b \dot{\hat{y}}_b = \dot{\hat{x}} \tag{5}$$

which clearly satisfy the object kinematic constraint.

The main idea behind proposing the above solution, is not only to establish a common tool-frame velocity (as required by the firm grasping constraints) but also to resort to the common tool-frame velocity as a reasonable compromise between two individual optimal values; in any case without requiring any unfeasible amount of information exchange between the agents. As a matter of fact, further investigations are needed for measuring the degree of sub-optimality introduced by the employment of $\dot{\hat{x}}$ in lieu of the globally optimal one $\dot{x}^\circ$, obtained by solving the problem in a centralized way. Indeed $\dot{x}^\circ$ remains, although almost impossible to be real-time evaluated within underwater environment, a reference stone with respect to which any suboptimal control law has to be compared. Moreover the possible existence of secondary effects, if any, that might appear as a consequence of the introduced sub-optimality, are still to be investigated. However, the extensive simulations performed, some of which are presented in the section 5, show encouraging results even by just using the arithmetic mean between $\dot{x}_a$, $\dot{x}_b$.

B. Compensation of Disturbances

In the above section we have presented the distributed algorithm for cooperating UVMS. In particular, the second independent optimization, having the end-effector as the first task, allows to satisfy also the kinematic constraint indirectly whenever both systems are given the same velocity to track. However, this does not hold anymore whenever velocity disturbances (or equivalently, velocity tracking errors) are considered. This is particularly crucial for the vehicles because they are more affected by sea currents and less accurate than the arms. This problem would inevitably create stress on the object due to the violation of the object kinematic constraint.

The idea to reduce the effects of vehicle disturbances is to modify the second optimization by adding, in parallel to the above procedure, an arm control law parametrized by the actual vehicle velocity. This can be done by considering the same task hierarchy of Section III-A3 but with just the arm velocities as control variables, and the vehicle velocity as a parameter. The output of this procedure is an arm control *law* for each UVMS:

$$\dot{\bar{q}}_a = \bar{\rho}_a + P_a v_a, \tag{6}$$
$$\dot{\bar{q}}_b = \bar{\rho}_b + P_b v_b, \tag{7}$$

where $\bar{\rho}_a$, $\bar{\rho}_b$ are the known terms output of the optimization procedure, and the P matrices express the influence of the
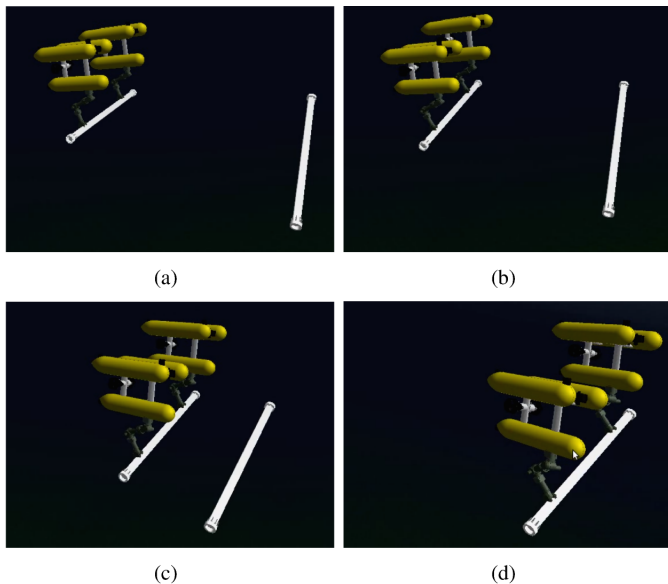
Fig. 2. Different stages in the transportation and positioning phase; the pipe on the bottom-right is the goal position to be reached

respective vehicle velocities on the arm velocity reference. We remark that the above arms control laws are individually the best ones in correspondence of *any* vehicle velocities v_a and v_b.

The computation of this arm control law is an additional effort w.r.t. the already computed system velocities $\dot{\hat{y}}_a$ and $\dot{\hat{y}}_b$ that naturally also encompass the arm velocities. We note that substituting $\hat{v}_a$ and $\hat{v}_b$ in (6) would lead to $\dot{\tilde{q}}_a = \dot{\hat{q}}_a$ (and the same for system b), because $\dot{\hat{q}}_a$ is the best arm velocity in correspondence of the generated vehicle reference $\hat{v}_a$. However, the substitution to be performed in (6) is to use the vehicle velocity *feedback*. This allows the arm to compensate for vehicle velocities tracking inaccuracies.

It is important to note that a perfect compensation is possible only if the arm end-effector Jacobian is full row rank, otherwise, even if the end-effector velocity task is the first one, there could be some particular directions where the compensation is not possible. For this reason, the manipulability task assumes even more importance, as it allows to maintain the Jacobian always well-defined. However, since the vehicle is affected by disturbances and the arm might not have enough DOFs left to execute the manipulability task, this cannot be guaranteed under all circumstances. Thus, the effects of disturbances on the overall system are still the subject of a deeper investigation, despite the preliminary simulations presented here are showing encouraging results.

IV. SIMULATION RESULTS

This section gives details about the preliminary simulation results obtained by using the algorithms described in the sections above. The goal of the simulation trials was to transport an object from an initial starting position to a final one, beginning the simulation from an initial condition where the object is grasped, using two free-floating vehicles (6 DOFs)
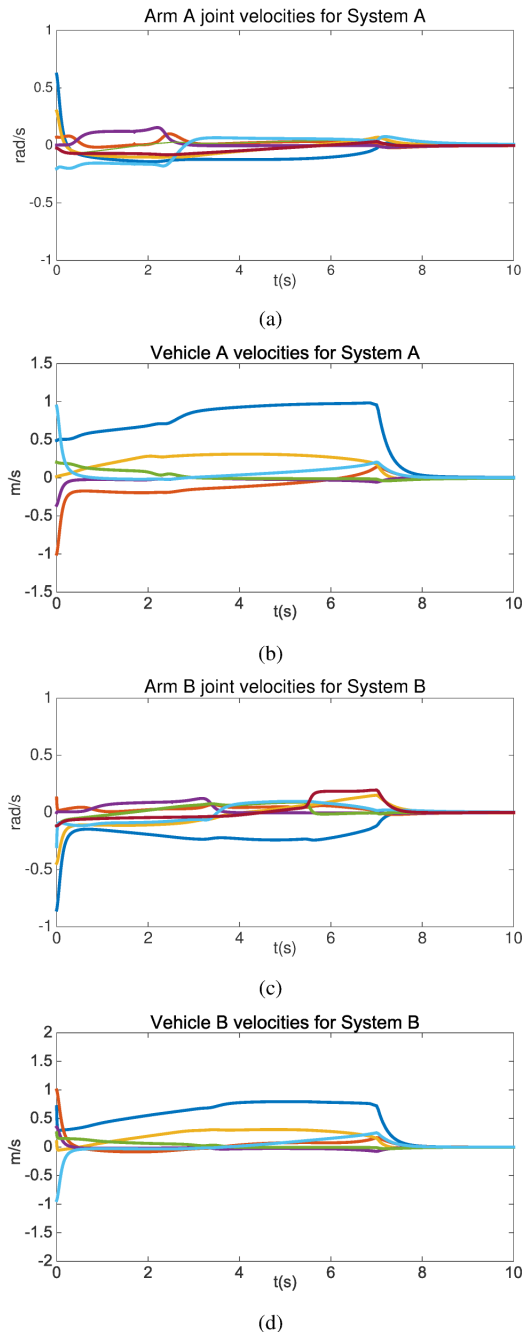

Fig. 3. First simulation: arm joint velocities and vehicle velocities of System A and B

endowed with two redundant manipulators (7 DOFs). The tasks used in this simulation are: keeping away from joint limits, keeping the manipulability measure above a certain threshold, maintaining the horizontal attitude of the vehicles, maintaining a fixed distance between the vehicles, reaching the desired goal position. The tasks related to the camera have not been included in these preliminary simulations and will be added in the future.

For these simulations we have used the activation functions as presented in [4], i.e. bell-shaped functions of the task vari-
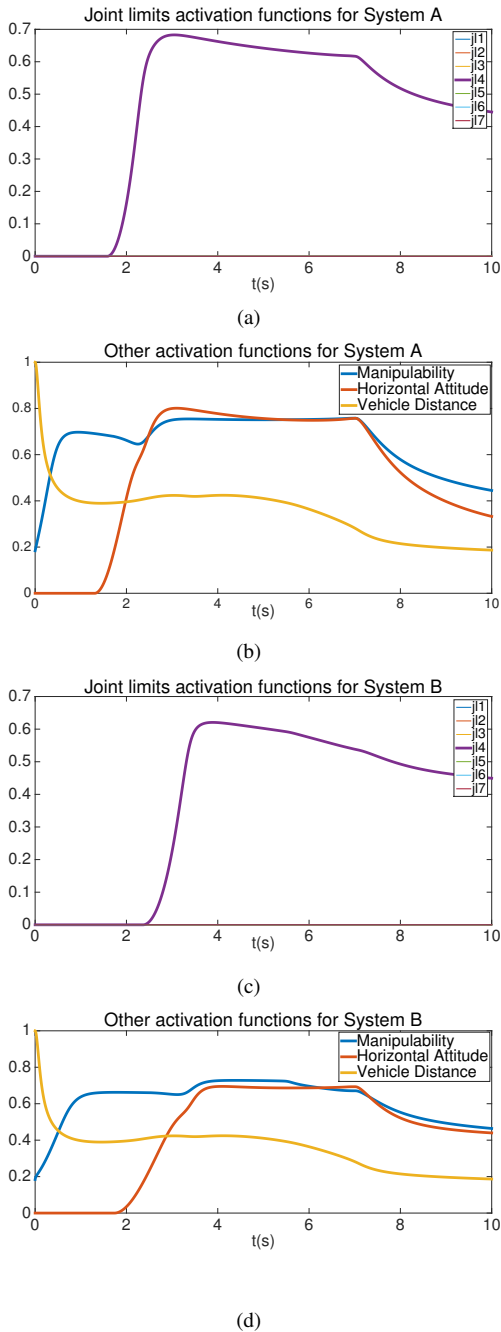
(a)

(b)

(c)

(d)

Fig. 4. First simulation: activation functions for System A and B

able with finite support. In particular, the activation function is equal to one whenever the corresponding inequality is not satisfied and it smoothly goes to zero whenever the variable is inside the region where the inequality is instead satisfied. The buffer zone where the activation function is greater than zero but the inequality is already satisfied serves the purpose of avoiding chattering phenomena.

In the first simulation, the two systems are commanded to transport the object 7 m along the x-axis, 3 m along the y-axis and -1 m along the z-axis. Therefore goal position of the object in (x, y, z) coordinates is (7,3,-1) meters rotated by an

angle of 30 degrees around the x-axis. UWSim [7], which is an underwater robotics simulator, has been used for performing visual rendering.

Figure 4 reports the time history of the activation functions for the joint limits, manipulability, horizontal attitude and distance between the vehicles tasks for the systems a and b. From figures 4(a), 4(b), 4(c), 4(d) it can be seen that different tasks are in transition during the trial simulations. The proposed framework allows to maintain the corresponding variable of each task within its given boundaries using suitable smooth activation functions for activating and deactivating the tasks. With this approach, the system manages to successfully accomplish the final objective of the mission, by transporting the object to the desired goal position as seen from the Fig. 2. The arm joint velocities and the vehicle velocities obtained during simulation for system a and system b in Fig. 3 shows the smoothness of the control that has been achieved.

In the second simulation the goal position is at (2,1,3) meters, rotated of an angle of 30 degrees around y axis and 45 around z axis. Furthermore, in this simulation a sinusoidal disturbance of 0.2m/s amplitude and 0.5Hz frequency is added to the vehicle velocities along the x axis of the inertial frame. Figure 5 shows the velocities of the two UVMS, while Fig. 6 reports the time history of their activation functions. The sinusoidal behaviors that can be seen in the generated velocities are consequence of the disturbance that is added to the vehicle velocity. Thanks to the proposed compensation approach, the two system successfully bring the tool-frame on top of the goal-frame despite the simulated sea current, as shown in Fig. 7.

V. CONCLUSION

In this paper we have presented a novel algorithm for the autonomous cooperative transportation of large objects by two free floating vehicles endowed with redundant manipulators. The algorithm is structured in such a way to minimize the information exchange between the two UVMSs, as they only need to exchange their respective tool-frame velocities. On going works are dedicated to further analysing the disturbances compensation technique. One of the major challenges now would be to completely avoid the exchange of information between the agents, by relying on the force/torque sensing at the wrist of both arms.

REFERENCES

[1] 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.
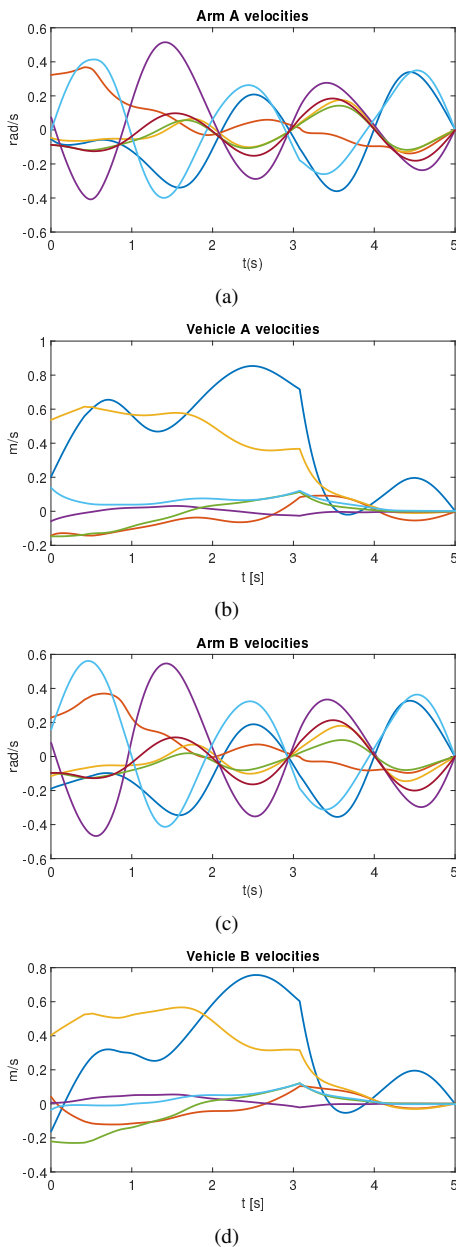
Fig. 5. Second simulation: arm joint velocities and vehicle velocities of System A and B
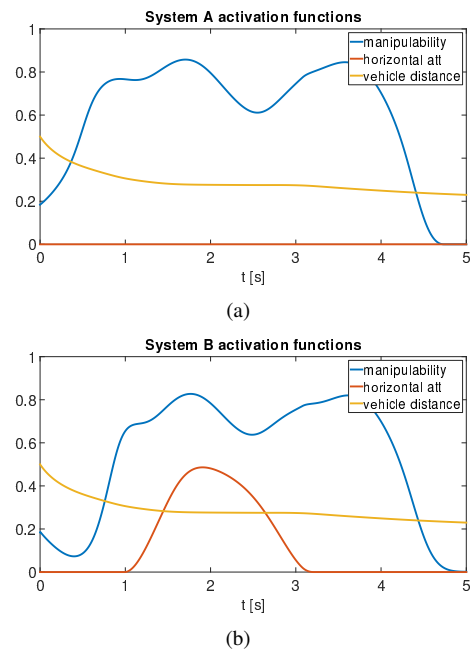

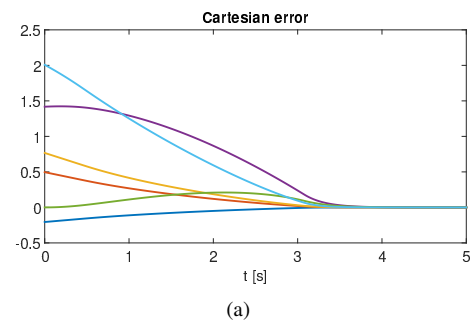Fig. 6. Second simulation: activation functions for System A and B

Fig. 7. Second simulation: Cartesian error between the tool and goal frame

control methodology and experimental validation within the trident project," *Journal of Field Robotics*, vol. 31, no. 3, pp. 364–385, May 2014.

[5] 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.

[6] 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.

[7] 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.

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

[3] 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.

[4] E. Simetti, G. Casalino, S. Torelli, A. Sperinde, and A. Turetta, "Floating underwater manipulation: Developed