# Adaptive Admittance Control in Task-Priority Framework for Contact Force Control in Autonomous Underwater Floating Manipulation*

Patryk Cieślak[1] and Pere Ridao[2]

***Abstract*— This paper presents a control architecture for an underwater vehicle-manipulator system (UVMS) to enable simultaneous tracking of end-effector configuration and contact force during floating-base manipulation. The main feature of the architecture is its combination of a task-priority (TP) kinematic control algorithm with a custom force control strategy, based on impedance (admittance) control. The TP algorithm used in the work includes recent treatment of equality and inequality tasks as well as original concepts to handle operation in singular configurations of the system. In the force control part the impedance concept is extended to allow for direct control over the value of exerted force and torque. Additional feed-forward signal is used to ensure stable contact. The performance of the control architecture is demonstrated by experiments in a test tank, with GIRONA500 I-AUV performing pipe inspection.**

## I. INTRODUCTION

Underwater intervention at sea remains an extremely costly and timely operation, with the use of large and heavy remotely operated vehicles (ROV) and special dynamic positioning (DP) vessels. In the last three decades, research in autonomous underwater robots and robotic intervention is slowly gaining speed. It all started with the pioneering works of OTTER, ODIN [6], UNION [18] and AMADEUS [12] but the first field demonstrations arrived during the ALIVE project [9], where an autonomous underwater vehicle (AUV) docked to a subsea intervention panel using hydraulic grasps and performed fixed-base valve turning. Similar experiments were later presented in the TRITON [16] project. First intervention from a floating vehicle was achieved in the SAUVIM [13] project. An intervention-AUV (I-AUV) autonomously located and hooked an object while hovering, to recover it from the seabed. First multipurpose object search and recovery strategy was demonstrated in the TRIDENT [21] project, with experiments both in the water tank and in the harbour environment. The vehicle used in the trials was GIRONA500, equipped with a 4 DOF arm or a 7 DOF manipulator with a 3 finger hand. Next, the PANDORA project has demonstrated autonomous floating-base valve turning on a subsea panel, first using a learning by demonstration (LbD) paradigm [4] and later utilising more standard task-priority (TP) kinematic control [7]. Finally, the MARIS project [23] has shown detection and grasping of a pipe on the seabed, with a specially designed 3 finger hand.

Above projects have focused on controlling the position of the robot end-effector (EE). Although some I-AUVs were equipped with force/torque sensors, they were only used to detect contact and aid in triggering actions [11], [23]. Moreover, all experimental trials involved grasping and recovering objects or manipulating valves and connectors. There is no example of a task which relies on accurate control of contact forces, like the presented here pipe inspection, highly relevant to the oil and gas industry. However, some theoretical works with simulated experiments can be found in the literature. In [8] a unified force control for an I-AUV is shown. The authors use dynamic model of the system and interleve impedance force control with hybrid position/force control, to allow tracking of the desired contact force and at the same time ensure soft behaviour on approach. The two strategies are switched with fuzzy logic. A similar approach was also presented in [2]. A different algorithm, based on passivity concept, is proposed in [15]. The dynamic model of an I-AUV is not needed by the controller and the kinematic constraints arising from the contact with environment are explicitly included when solving for system velocities. Authors present robustness of the algorithm to fluid disturbances and obstacle avoidance capabilities. In [10] a model based multiple impedance control is used to manipulate an object with a dual-arm I-AUV. Next, a force control algorithm, avoiding dynamic modelling, is shown in [5] for the task of valve turning and button pushing, simulated on a model of a 13 DOF I-AUV. Finally, simulated results of autonomous underwater pipe inspection, including contact force control, were shown in the context of the DexROV project [22].

In this paper, a control architecture is presented, to enable robust and safe underwater manipulation from a floating I-AUV, especially in the case where the interaction forces have to be controlled and the contact should be maintained during the whole operation. The paper is organised as follows. Section II presents the general control architecture of our I-AUV system, while Sections III and IV focus on EE force and position control parts respectively. These two sections present the main contributions of the paper (in bold). The proposed theory is put into practice in Section V, which presents preliminary experimental results of a pipe inspection operation, obtained with GIRONA500 I-AUV in a test tank. These results constitute, best to our knowledge, the first documented experimental example of autonomous underwater floating manipulation with contact force control. The work is concluded in Section VI.

*Patryk Cieślak has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement no. 750063.

[1]Patryk Cieślak is with the Computer Vision and Robotics Research Institute, Scientific and Technological Park of the University of Girona, CIRS lab, 17003 Girona, Spain patryk.cieslak@udg.edu

[2]Pere Ridao is with the Computer Vision and Robotics Research Institute, Scientific and Technological Park of the University of Girona, CIRS lab, 17003 Girona, Spain pere.ridao.rodriguez@gmail.com

## II. CONTROL ARCHITECTURE

The control architecture of the robot is schematically presented in Fig. 1. The trajectory generator outputs desired values of the EE configuration and wrench. The desired EE configuration is modified by the contact force control block, based on the measurements from the force/torque (FT) sensor located in the manipulator wrist. Then, it is passed to the kinematic control layer which implements the TP algorithm and generates velocities for the system to follow, in all controlled DOF. The velocity control layer is based on proportional-integral-differential (PID) controllers and feed-forward compensation of thruster dynamics. The manipulator has its own built-in velocity controllers. The thrust control layer is open-loop and implements nonlinear relationship between the requested wrench and the commanded thruster set-points. Moreover, there is a feed-forward wrench signal generated by the contact force control.

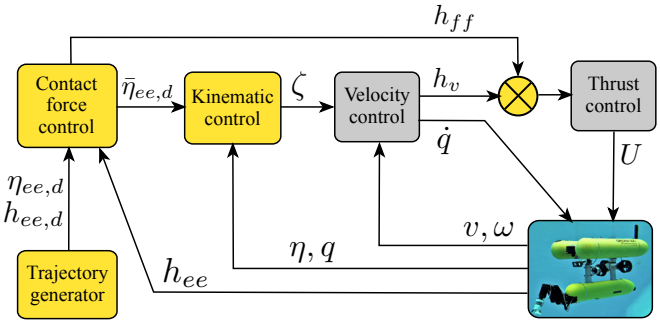


Fig. 1. Scheme of the control architecture of GIRONA500 I-AUV. The symbols used in the drawing are explained in the text. Work presented in this paper concerns the yellow coloured blocks.

## III. CONTACT FORCE CONTROL

### A. *Adaptive admittance control*

The contact force control layer is based on impedance control. In case of industrial manipulators, impedance control usually generates inputs to an inverse dynamics algorithm. However, in our case, it is separated from the motion control, which is model-free. This control scheme is often called the admittance control [19]. It is based on the concept of a *compliant frame* $\bar{\eta}_{ee,d}$ attached to the *stiff* desired EE frame $\eta_{ee,d}$ by a virtual linear mass-spring-damper system, with dynamics described by

$$\begin{aligned} M\ddot{\tilde{\eta}}_{ee} + K_D\dot{\tilde{\eta}}_{ee} + K_S\tilde{\eta}_{ee} &= h_{ee} \\ \bar{\eta}_{ee,d} &= \eta_{ee,d} + \tilde{\eta}_{ee}, \end{aligned} \tag{1}$$

where $M$, $K_D$, $K_S$ are impedance parameters (constant, diagonal matrices), $\tilde{\eta}_{ee}$ is the pose error between the stiff and the compliant EE frame, where $\eta = [\eta_1\ \eta_2]^T = [x\ y\ z \mid \phi\ \theta\ \psi]^T$, and $h_{ee} = \left[f_{ee}^T, \tau_{ee}^T\right]^T$ is the EE contact wrench, measured by the FT sensor installed in the robot wrist. The algorithm allows for a good control over the character of contact between the EE and the environment, securing the robot and the manipulated objects against damage and allowing for uncertainty in the estimation of the pose of both. The most obvious drawback of the algorithm is the lack of direct control over the contact wrench value. To fix it we replace the constant stiffness $K_S$ with a variable one, given by

$$K_S(t) = K_P \int \left(h_{ee,d}(t) - h_{ee}(t)\right) dt, \tag{2}$$

where $h_{ee,d}$ is the desired EE wrench and $K_P$ is a diagonal gain matrix. All impedance parameters, together with the initial $K_S$ and $K_P$ have to be tuned during experimental trials.

### B. *Feedforward force extension*

Due to the fact that our actual motion control is based on kinematic control, its performance is limited by the performance of the velocity control layer. Accurately tracking velocities is especially difficult with the I-AUV floating at all times, susceptible to noise, nonlinearities and delays in the navigation and the thrust control layer. Therefore, although in theory it is possible to track the desired force using only adaptive admittance, in reality we need an additional component which ensures that the EE is always pushing on the inspected surface, stabilising the system at the same time. One has to realise, that in the steady-state the contact force is always generated by the vehicle's thrusters. Even if the manipulator motion is the source of initial force impulse it is eventually necessary to counteract this impulse with the vehicle thrust. There is only a short transient period when the vehicle inertia and drag helps the manipulator to generate force. This observation led to the idea of generating feedforward force and torque components, passed directly to the vehicle thrust control layer. The robot is modelled as a rigid body that is in contact with the environment at the EE frame $\{E\}$ origin and the contact force is equal to the desired force vector. Gravity and buoyancy was not included in the model, i.e., non-zero vehicle pitch and roll angles do not generate contact force. The wrench that has to be generated by the thrusters is then simply given by

$$h_{ff} = \begin{bmatrix} {}^N R_B f_{ee,d} \\ {}^B r_{BE} \times \left({}^N R_B f_{ee,d}\right) \end{bmatrix}, \tag{3}$$

where ${}^N R_B$ is a rotation matrix from the world frame $\{N\}$ to the vehicle body frame $\{B\}$, aligned with robot centre of gravity (CG), $f_{ee,d}$ is the desired contact force and ${}^B r_{BE}$ is a vector from the origin of frame $\{B\}$ to the origin of the EE frame $\{E\}$, defined in frame $\{B\}$. Which elements of the feedforward torque can be generated depends on the number and configuration of the vehicle's thrusters. In reality, the thrust control is far from perfect and there is always a misalignment between EE frame and the surface normal vector, therefore only a fraction of (3) is used, just to maintain contact.

## IV. KINEMATIC CONTROL

The objective of the kinematic control is to find a vector of desired quasi-velocities $\zeta$ which will drive the EE at a desired velocity $\dot{x}_E$ to achieve a certain goal:

$$\dot{x}_E = J(\mathfrak{q})\zeta, \tag{4}$$

where $\dot{x}_E = \begin{bmatrix} v_E^T & \omega_E^T \end{bmatrix}^T$, $J(\mathfrak{q})$ constitutes the Jacobian matrix with $\mathfrak{q} = \begin{bmatrix} \eta^T & q^T \end{bmatrix}^T$, where $q$ is a vector of positions of manipulator joints, and $\zeta = \begin{bmatrix} \nu^T & \dot{q}^T \end{bmatrix}^T$. A solution to (4) is obtained by computing the inverse of the Jacobian matrix. In case of a redundant system, like an I-AUV, the Jacobian matrix is non-square so the Moore-Penrose pseudo inverse has to be used [14]:

$$\zeta = J^{\dagger}(\mathfrak{q})\dot{x}_E + \left(I - J^{\dagger}(\mathfrak{q})J(\mathfrak{q})\right) y, \tag{5}$$

where $y$ is an arbitrary vector and $\left(I - J^{\dagger}(\mathfrak{q})J(\mathfrak{q})\right) = P$ is called a null-space projector.

### A. Task-priority redundancy resolution

According to (5), in case of a redundant system, it is possible to generate velocities in the null space of the solution, which will not affect the main task, by defining the $y$ vector. In general, let us consider a system with $n$ controllable DOF and define a task variable $\dot{x}_1 \in \mathbb{R}^{m_1}$ related to its corresponding quasi-velocities vector by

$$\dot{x}_1 = J(\mathfrak{q})_1 \zeta, \tag{6}$$

where $J(\mathfrak{q})_1 \in \mathbb{R}^{m_1 \times n}$ is the Jacobian of the task.
Solving for $\zeta$ gives

$$\zeta = J_1^{\dagger}(\mathfrak{q})\dot{x}_1 + \left(I - J_1^{\dagger}(\mathfrak{q})J_1(\mathfrak{q})\right) y. \tag{7}$$

Let us also define a secondary task variable $\dot{x}_2 \in \mathbb{R}^{m_2}$, with its accompanying Jacobian $J_2(\mathfrak{q}) \in \mathbb{R}^{m_2 \times n}$ related by

$$\dot{x}_2 = J(\mathfrak{q})_2 \zeta. \tag{8}$$

We are now looking for the $y$ vector which will ensure completion of the secondary task without affecting the primary one. Substituting (7) in (8) gives

$$\dot{x}_2 = J_2(\mathfrak{q})J_1^{\dagger}(\mathfrak{q})\dot{x}_1 + J_2(\mathfrak{q})\left(I - J_1^{\dagger}(\mathfrak{q})J_1(\mathfrak{q})\right) y. \tag{9}$$

Minimising the secondary task error we get

$$\begin{aligned} y &= \hat{J}_2^{\dagger}(\mathfrak{q})(\dot{x}_2 - J_2(\mathfrak{q})J_1^{\dagger}(\mathfrak{q})\dot{x}_1 + \left(I - \hat{J}_2^{\dagger}(\mathfrak{q})\hat{J}_2(\mathfrak{q})\right) z \\ \hat{J}_2(\mathfrak{q}) &= J_2(\mathfrak{q})\left(I - J_1^{\dagger}(\mathfrak{q})J_1(\mathfrak{q})\right) \end{aligned} \tag{10}$$

Finally, substituting (10) in (7) yields

$$\begin{aligned} \zeta = J_1^{\dagger}(\mathfrak{q})\dot{x}_1 + \hat{J}_2^{\dagger}(\mathfrak{q})\left(\dot{x}_2 - J_2(\mathfrak{q})J_1^{\dagger}(\mathfrak{q})\dot{x}_1\right) + \\ + \left(I - \hat{J}_2^{\dagger}(\mathfrak{q})\hat{J}_2(\mathfrak{q})\right) z \end{aligned} \tag{11}$$

which constitutes the quasi velocity vector that the system should follow to achieve the primary and the secondary task, leaving room for additional tasks in the $z$ vector.

It is possible to derive a recursive formula for an arbitrary number of $N$ tasks, which is given by [3]

$$\zeta_i = \zeta_{i-1} + \bar{J}_i^{\dagger}(\mathfrak{q})\left(\dot{x}_i - J_i(\mathfrak{q})\zeta_{i-1}\right), \quad i = 1 \ldots N, \tag{12}$$

where

$$\bar{J}_i(\mathfrak{q}) = J_i(\mathfrak{q})P_{i-1}, \quad P_i = P_{i-1} - \bar{J}_i^{\dagger}(\mathfrak{q})\bar{J}_i(\mathfrak{q}). \tag{13}$$

The algorithm has to be initialised with

$$\zeta_0 = \mathbf{0}^n, \quad P_0 = I^{n \times n}. \tag{14}$$

### B. Task definition

A task $\sigma_i$ is defined as a generic m-dimensional variable depending on the generalised coordinates $\mathfrak{q}$ [1]:

$$\sigma_i = \sigma_i(\mathfrak{q}) \in \mathbb{R}^{m_i} \tag{15}$$

It is connected with the task velocity vector $\dot{x}_i$ by

$$\dot{x}_i = \dot{\sigma}_i + K_i \tilde{\sigma}_i, \tag{16}$$

where $K_i$ is a diagonal gain matrix and $\tilde{\sigma}_i = \sigma_{i,d} - \sigma_i$. Looking at (16) we see two components: a feed-forward part $\dot{\sigma}_i$, which may be used, e.g., in case of tracking of a moving object, and a feedback part $K_i\tilde{\sigma}_i$ which drives the system to the goal $\sigma_{i,d}$.

### C. Inequality tasks

The TP algorithm in its pure form has one serious drawback. It would be natural to put safety tasks, e.g., manipulator joint limits, at the top of the hierarchy, above the tasks responsible for reaching the goal. However, in practice, it results in over-constraining the system so the goal cannot be achieved. To solve this problem, the inequality tasks (or set tasks) were introduced [20]. To fulfil an inequality task some variables have to be kept in a specified range rather than approach a specific value. As a result, these kind of tasks require a definition of a binary activation function $a_i(\mathfrak{q}) \in \{0, 1\}$. Then, the task hierarchy is continuously updated, depending on the values of the activation functions. Care has to be taken when switching the tasks though. For example, a joint limit task may be inactive according to the current joint value but the desired joint velocity computed by the TP algorithm will lead to the violation of the limit before the next update. Therefore, the quasi-velocities should be integrated based on the controller sampling time and if the violation is detected, then the joint limit task should be activated and the TP algorithm recomputed.

### D. Task hierarchy

*1) Task 1: Manipulator joint limits (inequality):* Ensuring that the control system generates velocities compliant with the manipulator joint limits is a typical safety task put on top of the task hierarchy. The task is defined for each joint $j$ as follows (written for upper limit $q_{j,1}$):

$$\tilde{\sigma}_i = q_{j,1} - \delta - q_j, \quad \dot{\sigma}_i = 0, \quad J_i(\mathfrak{q}) = J_{q,j}, \tag{17}$$

where $J_{q,j}$ is a single-entry row matrix, equal 1 for the entry corresponding to the joint $j$. As an inequality task it is characterised by the following activation function:

$$a_i = \begin{cases} 1, & a_i = 0 \wedge q_j \geq q_{j,1} - \alpha \\ 0, & a_i = 1 \wedge q_j \leq q_{j,1} - \delta \end{cases}, \tag{18}$$

where $\alpha$ and $\delta$ are the activation and deactivation margins respectively. It is required that $\delta > \alpha$ to avoid chatter.

*2) Task 2: Nominal value (vehicle yaw, equality):* Keeping a specific vehicle yaw is often desired to ensure good visibility of the target. Moreover, it automatically results in having the manipulator in front of the vehicle, where the EE is well visible and the manipulator is not generating excessive asymmetrical drag. Finally, when contact with the environment is required, keeping the EE close to the vehicle longitudinal axis produces the lowest roll torque, which is often only passively compensated for. The task variables are defined as follows:

$$\tilde{\sigma}_i = \psi_d - \psi, \quad \dot{\sigma}_i = \dot{\psi}_d, \quad J_i(\mathfrak{q}) = J_\psi, \tag{19}$$

where $J_\psi$ is a single-entry row matrix, equal 1 for the entry corresponding to $\psi$.

*3) Task 3: End-effector configuration (equality):* Finally, the most typical task used to follow the mission plan is tracking the desired EE position and orientation. The task variables are defined as follows:

$$\begin{aligned} \tilde{\sigma}_i &= \eta_{ee,d} - \eta_{ee} = \begin{bmatrix} \eta_{1_{ee,d}} - \eta_{1_{ee}} \\ \lambda_{ee}\varepsilon_d - \lambda_d\varepsilon_{ee} + S(\varepsilon_d)\varepsilon_{ee} \end{bmatrix} \\ \dot{\sigma}_i &= \dot{x}_{E,d}, \quad J_i(\mathfrak{q}) = J(\mathfrak{q}), \end{aligned} \tag{20}$$

where the orientation error was computed in quaternion sense and $S(\cdot)$ is a cross-product matrix operator [1]. This representation does not suffer from gimbal-lock problems.

### E. Operation in singular configurations

Depending on the kinematic chain of the I-AUV and the task to be performed, it may happen that the robot has to work in a singular configuration where two or more axes of the robot DOF align. This situation can be detected by monitoring the measure of manipulability (MOM)

$$m = \sqrt{\det(JJ^T)}. \tag{21}$$

A special task can be defined for highly redundant systems to keep MOM above a predefined threshold. However, it is sometimes not possible to avoid working in singular configurations, like in case of our 8 DOF I-AUV during pipe inspection. Without a special treatment of this situation the robot may get stuck around singular configurations because TP generates high quasi-velocities in the aligned joints, frequently changing directions. To lower the resulting quasi-velocities the weighted damped least-squares (DLS) solution to computing the Jacobian inverse is used [7]. However, it does not ensure that the system will not get stuck, so we propose an additional algorithm to selectively lock DOF just before the main (goal) task in the hierarchy, which is commonly the EE configuration task. Locking means injecting additional *nominal tasks* into the hierarchy which define the desired position of the selected DOF equal to the position at the time of locking. This redundancy reduction algorithm engages only in the vicinity of singular configurations thanks to the continuous monitoring of the main task Jacobian matrix $J(\mathfrak{q}) \in \mathbb{R}^{m \times n}$ and works according to Algorithm 1, where $\varepsilon \approx 0.95$, *Sort()* is a sorting function which orders elements of $A$ in an ascending manner, *InjectTask()* injects tasks to the TP hierarchy $\mathcal{H}$, before the main task, and *Cost()* is a cost function letting us express the preference of locking specific DOF. One logical choice is to construct a function which will prefer locking manipulator DOF, and the ones that are closest to the limits, i.e., with least available range of motion, which can be formulated as

**Algorithm 1** Automatic redundancy reduction

```
for k ← 1, m do
    A ← ∅
    for j ← 1, n do
        if J[k, j] > ε then
            c ← COST(j)
            A ← A ∪ {(c, j)}
        end if
    end for
    if |A| > 1 then
        SORT(A)
        for j ← 1, |A| − 1 do
            INJECTTASK(A_j, H)
        end for
    end if
end for
```

$$c(j) = \begin{cases} \infty & \text{vehicle DOF} \\ -\frac{|(\mathfrak{q}_j - (\mathfrak{q}_{j,1} + \mathfrak{q}_{j,0})/2)|}{(\mathfrak{q}_{j,1} - \mathfrak{q}_{j,0})/2} & \text{otherwise} \end{cases}, \tag{22}$$

where $\mathfrak{q}_{j,0}$ and $\mathfrak{q}_{j,1}$ denote lower and upper limits of DOF $j$ respectively. As a side effect this criterion effectively ensures compliancy with manipulator limits in most situations, so that *manipulator joint limits task* engages much less often.

## V. EXPERIMENT

### A. Experimental setup

The developed control algorithm was implemented on-board the GIRONA500 I-AUV and tested in a pipe inspection scenario, in the CIRS test tank. GIRONA500 is a compact-sized underwater robot designed and developed at the University of Girona [17], which can be adapted to different tasks by changing its payload and number of thrusters. Here the vehicle was used in a 4 DOF configuration (yaw, surge, sway, heave) and equipped with an electric ECA 5E Micro manipulator. The vehicle is passively stable in pitch and roll and the manipulator has 4 DOF actuated by screw drives, which make it powerful but slow. Moreover, the drives exhibit high friction forces resulting in velocity dead-band. The vehicle is equipped with two vertical thrusters which enable generating a torque in pitch, used in $h_{ff}$. The kinematic structure of the system is presented in Fig. 2.

An aluminium structure was built to support a 2 m section of a 500 mm PVC pipe. The pipe was placed high enough above the bottom to ensure correct operation of the Doppler velocity log (DVL), being the most important navigation sensor. Arrays of four markers (ARUCO) were placed on both ends of the pipe to facilitate its stable detection with an on-board Point Grey Bumblebee 2 camera. The test rig is presented in Fig. 3. The vehicle was equipped with a strain

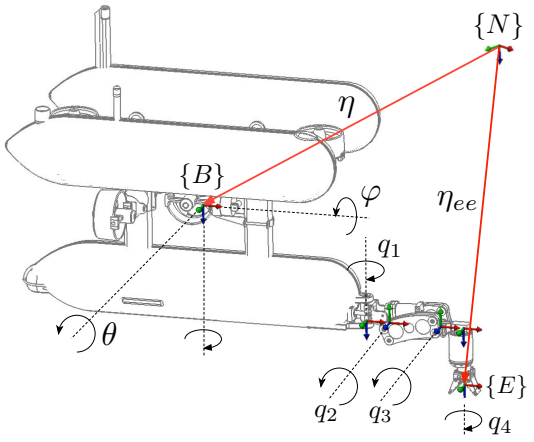


Fig. 2. Kinematic structure of GIRONA500 I-AUV.

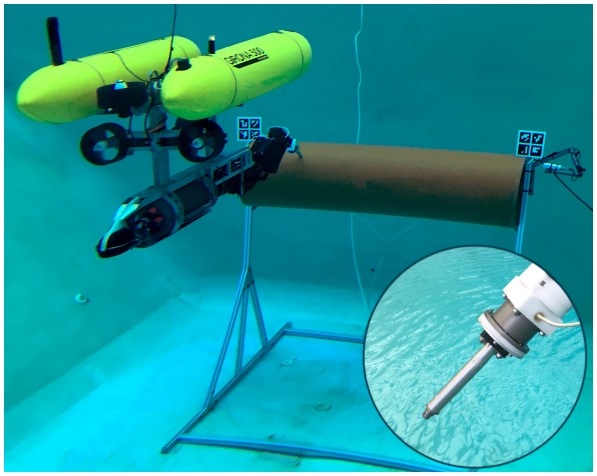

Fig. 3. GIRONA500 I-AUV performing pipe inspection in the test tank. Also shown is the end-effector equipped with a force/torque sensor.

gauge 6-axis force/torque sensor in the manipulator wrist and a simple end-effector with a cylindrical, flat-headed, rigid tip.

### B. *Path generation*

Desired EE path is computed based on the detected pipe axis, central point and a vertical axis aligned with marker surface. The user specifies a pice-wise-linear path in cylindrical coordinates $(l_p, \varphi_p)$, where $l_p$ is the translation along the pipe axis from the central point and $\varphi_p$ is an angle between the vertical axis and a line parallel to the pipe radius, passing through a desired point on pipe surface. Desired EE orientation is computed based on surface normal at each point of the generated path.

### C. *Experimental results*

The results of the pipe inspection experiment are presented in Fig. 4-8. The I-AUV was commanded to autonomously follow a rectangular path in cylindrical coordinates, including motions around and along the pipe, see Fig. 4. Due to pipe detection errors the path was generated assuming a pipe diameter which was significantly smaller than the real one, hence ensuring that the contact will always occur. Moreover,

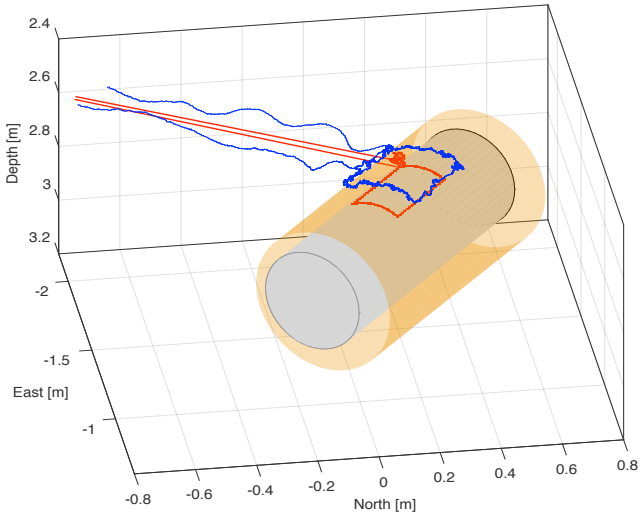


Fig. 4. Path of end-effector during pipe inspection. Shown are the stiff desired path (red), computed based on pipe detection with a pipe diameter set to 350 mm, and the actual path (blue), generated by the contact with the real pipe of diameter 500 mm. The two pipes do not coincide because there is a detection error, mainly in the distance between robot and pipe.

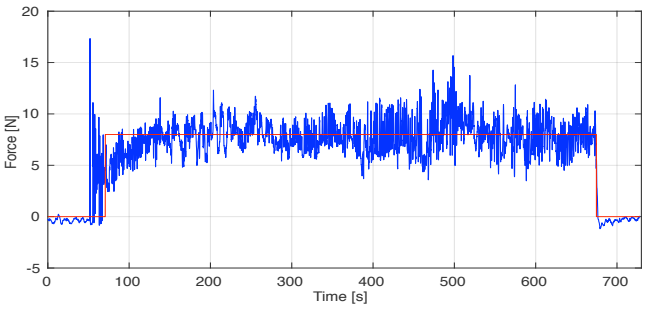


Fig. 5. Tracking of the desired contact force, along z-axis of $\{E\}$. Red line denotes the desired force and blue is the measured force.

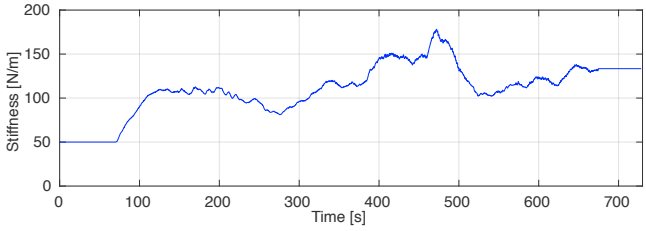


Fig. 6. Value of the variable stiffness in the impedance model. In this case all diagonal entries of $K_S$ were set to the same value.

the area around the top of the pipe was selected on purpose, to show how the automatic redundancy reduction kicks in when multiple system axes align, see Fiq. 7. Contact between the tip of the end-effector and the surface of the pipe was kept properly while following the path, as shown in Fig. 5. The impact behaviour can be observed just before the desired force is set. At this stage the admittance model uses default parameters. When the system stabilised, the desired force was set to 8 N and the automatic stiffness adjustment got engaged, see Fig. 6. Looking at Fig. 8 it can be noticed that

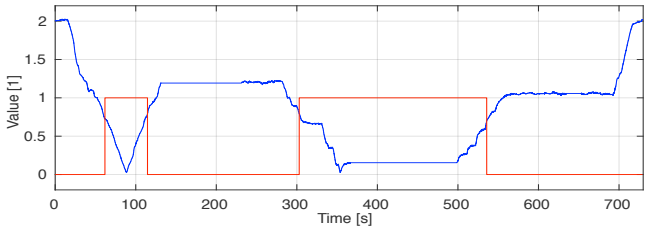


Fig. 7. Functioning of the automatic redundancy reduction algorithm. Blue line is the value of MOM and red line is the number of locked DOF.

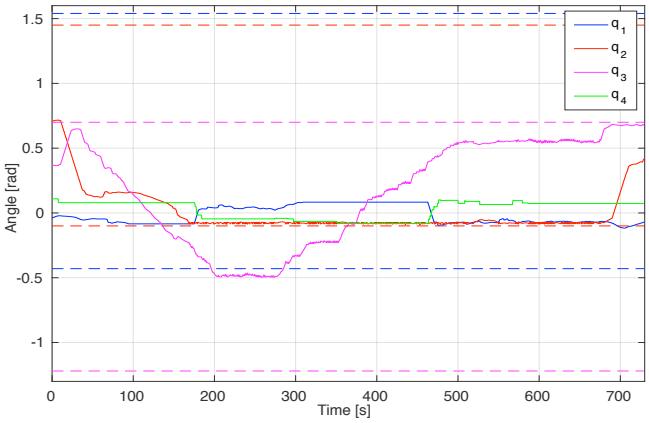


Fig. 8. Compliancy with the manipulator joint limits (inequality task). Continuous lines denote joint positions, dashed lines represent limits.

the joint movements were not smooth. This effect occurs due to the fact that the path is followed point by point with some resolution, the joint limits get activated and the manipulator exhibits velocity dead-band. The overall performance of the system is severely limited by the used manipulator and a poor control of vehicle velocity in heave.

## VI. CONCLUSIONS

The paper presents a control system for an I-AUV to enable contact force control during intervention missions which require it. The control ideas are based on admittance control combined with a modern task-priority framework. The admittance control was modified to allow direct control over the contact force value by introducing variable, self-adjusting stiffness matrix. Moreover, a feed-forward force is generated by the algorithm to maintain contact in case of velocity controller instability during collision. There is also an original idea of a redundancy reduction algorithm described. It enabled correct system operation in singular configurations, often encountered when using our 8 DOF I-AUV. The theory is proven by the results of a successful experimental trial representing a pipe inspection operation.

## ACKNOWLEDGMENT

This work is part of a project titled "Force/position control system to enable compliant manipulation from a floating I-AUV", which received funding from the European Union's Horizon 2020 research and innovation programme, under the Marie Sklodowska-Curie grant agreement no. 750063.

## REFERENCES

[1] G Antonelli. *Underwater Robots*, volume 96 of *Springer Tracts in Advanced Robotics*. Springer, Cham, December 2013.

[2] M R Arshad, S M Suboh, I A Rahman, and M N Mahyuddin. Fuzzy Logic Approach for Hybrid Position/Force Control On Underwater Manipulator. In *Proceedings of Malaysia-Japan International Symposium On Advanced Technology*, November 2007.

[3] P Baerlocher and R Boulic. Task-priority formulations for the kinematic control of highly redundant articulated structures. In *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, 1998.

[4] A Carrera, N Palomeras, N Hurtos, P Kormushev, and M Carreras. Learning by demonstration applied to underwater intervention. In *Proceedings of the Seventeenth International Conference of the Catalan Association of Artificial Intelligence (CCIA)*, 2014.

[5] E Cataldi and G Antonelli. Basic interaction operations for an underwater vehicle-manipulator system. In *Proceedings of 17th International Conference on Advanced Robotics*, October 2015.

[6] S K Choi, G Y Takashige, and J Yuh. Experimental study on an underwater robotic vehicle: ODIN. In *Proceedings of IEEE Symposium on Autonomous Underwater Vehicle Technology*, December 1994.

[7] P Cieslak, P Ridao, and M Giergiel. Autonomous underwater panel operation by GIRONA500 UVMS: A practical approach to autonomous underwater manipulation. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2015.

[8] Yong Cui and N Sarkar. A unified force control approach to autonomous underwater manipulation. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2000.

[9] J C Evans, P Redmond, C Plakas, K Hamilton, and D Lane. Autonomous docking for Intervention-AUVs using sonar and video-based real-time 3D pose estimation. In *Proceedings of OCEANS 2003*, 2003.

[10] H Farivarnejad and S Ali A Moosavian. Multiple Impedance Control for object manipulation by a dual arm underwater vehicle–manipulator system. *Ocean Engineering*, 89(C):82–98, October 2014.

[11] N Jamali, P Kormushev, A C Viñas, M Carreras, and D G Caldwell. Underwater robot-object contact perception using machine learning on force/torque sensor feedback. In *Proceedings of IEEE International Conference on Robotics and Automation*, January 2015.

[12] D M Lane, JBC Davies, and G Casalino. AMADEUS: Advanced MAnipulation for DEep Underwater Sampling. *IEEE Robotics & Automation Magazine*, 4(4):34–45, December 1997.

[13] G Marani and J Yuh. *Introduction to Autonomous Manipulation, Case Study with an Underwater Robot, SAUVIM*, volume 102. Springer, Berlin, April 2014.

[14] Y Nakamura. Advanced robotics: redundancy and optimization, 1990.

[15] E Olguin-Diaz and G Arechavaleta. A passivity-based model-free force–motion control of underwater vehicle-manipulator systems. *IEEE Transactions on Robotics*, December 2013.

[16] N Palomeras, P Ridao, D Ribas, and G Vallicrosa. Autonomous I-AUV Docking for Fixed-base Manipulation. *IFAC Proceedings Volumes*, 47(3):12160–12165, January 2014.

[17] D Ribas, P Ridao, Lluís Magí, N Palomeras, and M Carreras. The Girona 500, a multipurpose autonomous underwater vehicle. In *Proceedings of OCEANS 2011*, April 2011.

[18] V Rigaud, E Coste-Maniere, and M J Aldon. UNION: underwater intelligent operation and navigation. *IEEE Robotics & Automation Magazine*, 5(1):25–35, March 1998.

[19] B Siciliano, L Sciavicco, L Villani, and G Oriolo. *Robotics*. Modelling, Planning and Control. Springer Science & Business, London, 2009.

[20] 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*, December 2016.

[21] E Simetti, G Casalino, S Torelli, A Sperind, and A Turetta. Floating Underwater Manipulation: Developed Control Methodology and Experimental Validation within the TRIDENT Project. *Journal of Field Robotics*, 31(3):364–385, 2014.

[22] E Simetti, S Galeano, and G Casalino. Underwater Vehicle Manipulator Systems: Control Methodologies for Inspection and Maintenance Tasks. In *Proceedings of OCEANS 2016*, May 2016.

[23] E Simetti, F Wanderlingh, S Torelli, M Bibuli, A Odetti, G Bruzzone, D L Rizzini, J Aleotti, G Palli, L Moriello, and U Scarcia. Autonomous Underwater Intervention: Experimental Results of the MARIS Project. *IEEE Journal of Oceanic Engineering*, pages 1–20, September 2017.