Controlling Ocean One

Gerald Brantner and Oussama Khatib

Abstract Using robots to explore venues that are beyond human reach has been a longstanding aspiration of scientists and expeditionists alike. The deep sea exemplifies such an unchartered environment that is currently inaccessible to humans. Ocean One (O_2) is an anthropomorphic underwater robot, designed to operate in deep aquatic conditions and equipped with an array of sensor modalities. Central to the O_2 concept is a human interface that connects the robot and human operator through haptics and vision. In this paper, we focus on O_2's control architecture and show how it enables an avatar-like synergy between the robot and human pilot. We establish functional autonomy by resolving kinematic and actuation redundancy, allowing the pilot to control O_2 in a lower-dimensional space. We illustrate O_2's hierarchical whole-body control tasks including manipulation and posture tasks, feed-forward compensation as well as constraint handling. We also describe how to coordinate the dynamics of body and arms to achieve superior performance in contact and demonstrate O_2's capabilities in simulation, experiments in the pool as well as deployment to its archeological maiden mission to the 'Lune', a French naval vessel that sunk to 91 m depth in 1664 in the mediterranean sea.

1 Introduction

Over the past decades, advances in automation have enabled robots to replace humans in performing manual labor [1, 2]. This was possible because the manufacturing floor is tightly structured and tasks are highly repetitive. The next evolution for robots is

G. Brantner (✉) · O. Khatib
Stanford Robotics Laboratory, 353 Serra Mall, Gates Computer Science,
Stanford, CA 94305, USA
e-mail: geraldb@cs.stanford.edu

O. Khatib
e-mail: ok@cs.stanford.edu

to proxy for humans in unstructured, inhospitable environments and advancing the boundaries of human exploration by entering places that are currently inaccessible. The deep sea exemplifies an environment that is inhospitable and largely inaccessible to humans. The field of ROVs has recently brought major advancements to underwater robots that can navigate, observe and map [3–5] and the need for underwater operations has led to the development of submersible manipulators [6, 7]. The O_2 concept offers the capability to perform operations typical for human divers by synthesizing a humanoid robot that is functionally autonomous with a human pilot, who provides higher-lever cognitive abilities, perception and decision making. In this paper, we focus on O_2's control architecture. We illustrate how the robot acquires functional autonomy in coordinating manipulation tasks with posture and constraints in a hierarchical manner. Subsequently we establish an avatar-like synergy by interfacing the human pilot with the robot through vision and bimanual haptic devices. We demonstrate O_2's capabilities in simulation, experiments at the pool and eventually its maiden deployment, where it explored a french naval vessel that sank to 91 m depth in the Mediterranean sea, and retrieved archeological artifacts (Fig. 1).
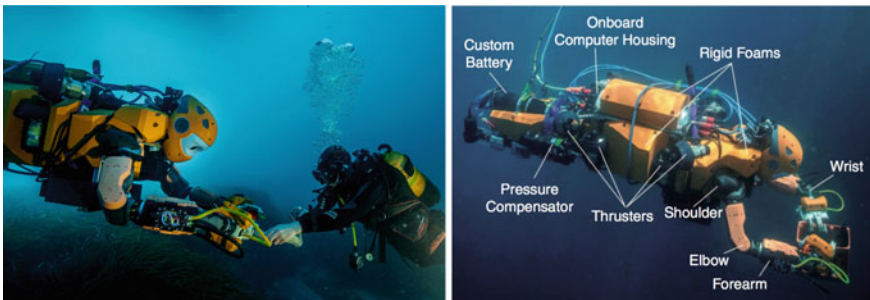

Fig. 1 Left: Ocean One interacting with a human diver. The human pilot—located on the research vessel *Andre Malraux*—interfaces with the robot through bimanual haptic devices and direct stereoscopic vision from the cameras mounted in its head. The robot mimics the pilot's hands while the pilot feels the forces perceived at the robot's hands. Right: Ocean One System Components. O_2 is a humanoid underwater robot with a body, two arms and a head. The series elastic arms are torque controlled and oil-filled. Each hand has three under-actuated tendon-driven fingers. O_2 has a pair of stereo cameras in the head and a wide-angle camera on the chest. The body is actuated by eight thrusters

2 Robot

Ocean One is a humanoid underwater robot of approximately adult-human dimensions and 200 kg overall weight. Its body is conceived in an anthropomorphic shape with shoulders, two arms and a head. Each arm has 7-DoF with electrically driven, torque-controlled joints adapted from the original Meka arm design. The arms are fitted with series elastic actuators that provide torque feedback to enhance compliant motion as well as force control, and safety. In order to withstand the pressure at oceanic depths, the arms are oil-filled and positively pressurized by spring-loaded compensators. Each hand has three fingers with three phalanges per finger that are driven with a tendon, which attaches to the distal finger phalanx, passes through the medial and proximal phalanges and loops around an axis driven by the single actuator [8]. The head contains a pair of high-definition cameras with global shutters. Pan and tilt actuation at the neck is currently being implemented. Another camera is attached on O_2's chest, offering a wide-angle perspective on the surroundings in front and below. The body is actuated by eight thrusters. More details on hardware components can be reviewed in [9].

3 Pilot Interface

The O_2 concept is not only the underwater robot itself but a distributed system of hardware and software components. The surface station allows the human pilot to connect to a set of interfaces: Haptic devices, GUI, live vision, and world display. These interfaces play different roles in different modes of operation. In Avatar-mode, the haptic devices and live vision are central, while GUI and virtualization are more predominant in autonomous modes (Fig. 2).

Fig. 2 Pilot Interface. The pilot's interaction with O_2 is established through an immersive interface consisting of bimanual haptic devices, stereoscopic live vision, GUI and a world display giving a global perspective of the scene

4   Modeling

O_2 is modeled as a tree-like structure, with two arms branching out from the body. We model its kinematics using generalized coordinates, with 6 virtual DoF for the body and 7 DoF for each arm. Each link is a rigid body with associated mass properties. This leads us to a multi-body dynamic system represented by twenty-dimensional equations of motion

$$A(q)\ddot{q} + b(q,\dot{q}) + g(q) + h = \Gamma, \tag{1}$$

where q is the vector of generalized coordinates, $A(q)$ is the kinetic energy matrix, $b(q,\dot{q})$ is the vector of centrifugal and coriolis forces, $g(q)$ the gravitational forces, h the hydrodynamic contribution and Γ is the vector of generalized forces. We extract the mass properties of the body and arms from the CAD models and experimental inspection.

5   Control

O_2 is a force controlled robot, which allows us to take advantage of the robot's dynamics at a global level, coordinating the slow dynamics of the body with the fast dynamics of the arms in order to achieve superior performance in motion and force control.

5.1   Whole Body Architecture

The objective behind O_2's control architecture is to enable a connection to the human interface, where a small set of control inputs is sufficient to pilot the entire robot, while achieving a high degree of autonomy already at the controller level. Because the interaction between robot and environment happens primarily through physical contact, we directly control the two hands while the body autonomously follows the hands. Beyond this, the controller monitors the pilot, the robot and its environment, and overrides actions that would lead to constraint violations, such as collisions with obstacles or joint limits. The remaining null-space is used to optimize arm and body posture for a given task. This behavior is realized by the control law

$$\Gamma = \underbrace{(J_c^\mathsf{T} F_c)}_{\text{Constraints}} + \underbrace{(J_{t|c}^\mathsf{T} F_{t|c})}_{\text{Manipulation}} + \underbrace{(J_{p|t|c}^\mathsf{T} F_{p|t|c})}_{\text{Posture}} + \underbrace{(J_h^\mathsf{T} F_h)}_{\text{Compensation}} , \tag{2}$$

a hierarchical architecture comprised of tasks. The four additive terms in (2) are associated with the contributions of *Constraints*, *Manipulation Task*, *Posture* and *feed-forward compensation*, respectively. The controller coordinates these tasks in

Fig. 3 Whole-Body control architecture. A hierarchical structure allows for prioritization of manipulation task, posture and constraints. The only control inputs are motion and forces of the left and right hand. Joint-limit, self-collision and obstacle-avoidance have highest priority. Posture is defined as the remaining controllable sub-space after constraints and manipulation-task. For gravitational and hydrodynamic compensation, the controller has an additional feed-forward term
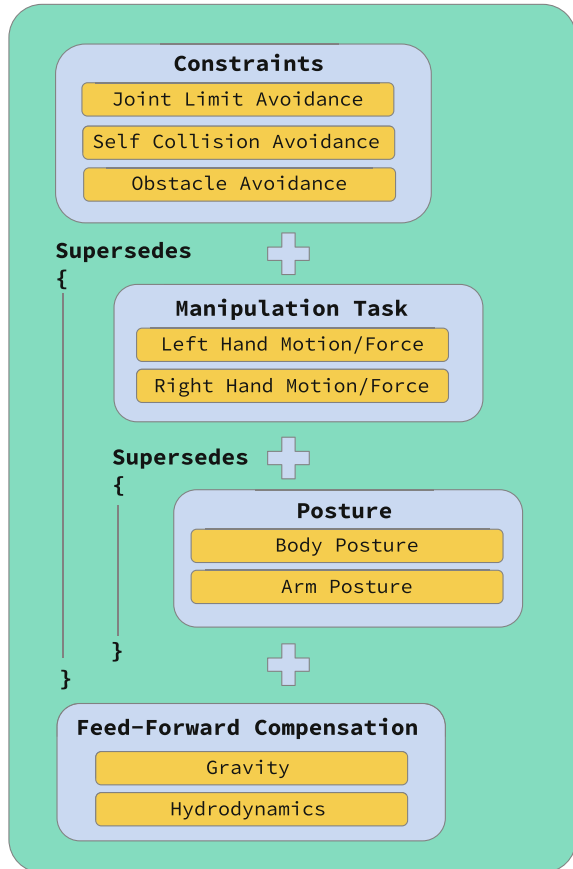
a prioritized manner. The notation $t|c$, for instance, reads *Task t consistent with Task c*. In case tasks are not simultaneously feasible, a lower-priority task will only be executed to the extent that is does not interfere with a higher-priority task. For instance, a new position goal at the hands might lead the body to collide with a rock. In such a case the constraint task will engage, and make the body evade the obstacle while still performing the task (Fig. 3).

For each task t, we specify an operational-space ϑ_t and a control force F_t. The task Jacobian J_t, establishes a dual velocity-force mapping between generalized coordinate space q to operational space ϑ with

$$\vartheta_t = J_t \dot{q} \tag{3}$$
$$\Gamma = J_t^\mathsf{T} F_t. \tag{4}$$

To prevent a lower-priority task from interfering with a higher-priority task, we filter Jacobians and control forces through null-space projections. Details on this implementation can be seen in [10].

5.2 Manipulation Task

The central element of the controller is the Manipulation Task, directly programming the hands. It is the only task that takes direct control inputs. These inputs are specified by six-dimensional operational space coordinates for each arm.

$$\vartheta_t^\mathsf{T} = [v_{\text{Left}}^\mathsf{T}, \omega_{\text{Left}}^\mathsf{T}, v_{\text{Right}}^\mathsf{T}, \omega_{\text{Right}}^\mathsf{T}] \tag{5}$$

$$\vartheta_t = J_t \dot{q} \tag{6}$$

v and ω represent the linear and angular velocities of O_2's hands. In this space we implement unified motion and force control laws expressed in F_t [11]. This abstraction also allows us to specify the manipulation task in a way that is agnostic of the robot's morphology.

5.3 Posture Task

After specifying the twelve-dimensional manipulation task, there are (in general) 8 uncontrolled DoF left. This remaining null-space (Fig. 4) is occupied by the posture task, which consists of two sub-tasks *Body Posture* and *Arm Posture*. *Body Posture* positions the body relative to the hands. Its goal is to align the body's coronal plane with the horizontal plane while aligning its longitudinal axis with the horizontal perpendicular to the axis connecting the two operational points and keeping a constant linear offset to the mid-point of the hands. This sub-task occupies 6 DoF.

$$\vartheta_{p,\text{Body}}^\mathsf{T} = [v_{\text{Body}}^\mathsf{T}, \omega_{\text{Body}}^\mathsf{T}] \tag{7}$$

$$\vartheta_{p,\text{Body}} = J_{p,\text{Body}} \dot{q} \tag{8}$$

Finally, there is 1 DoF of null-space left in each of the arms, which is controlled by the *Arm Posture* sub-task.

$$\vartheta_{p,\text{Arms}}^\mathsf{T} = [\dot{q}_{\text{Left}}^\mathsf{T}, \dot{q}_{\text{Right}}^\mathsf{T}] \tag{9}$$

$$\vartheta_{p,\text{Arms}} = J_{p,\text{Arms}} \dot{q} \tag{10}$$

Fig. 4 Left: Task specification. Every task is associated with an operational-space ϑ_{Task}, illustrated by the cartesian coordinate frames. The manipulation task consists of ϑ_{L} and ϑ_{R}, the six-dimensional spaces of the left and right hand—three coordinates for the frame origin (purple sphere) and three for the frame's orientation. These spaces are controlled directly by the manipulation task, which is the only task receiving control inputs. This abstraction allows for unified motion and force control that is agnostic of the robot's morphology. The body's operational space ϑ_{Body} is controlled by the Posture task such that it autonomously follows the hands. Right: Null-space of the manipulation task. The kinematics of O_2 are modeled with 20 DoF. The manipulation task occupies 6 DoF for each of the two arms. The remaining eight are controlled by the posture tasks. Here, O_2 performs a manipulation task on a red valve. The null-space is illustrated by various postures that leave the manipulation task variables (position and orientation of the white hands) undisturbed. This eight-dimensional posture can be used to execute the task in an optimized manner

In order to merge the two sub-tasks into a combined posture task, we stack the Jacobians $J_{p,\,\text{Body}}$ and $J_{p,\,\text{Arms}}$ and receive

$$\vartheta_p^{\mathsf{T}} = [\vartheta_{p,\text{Body}}^{\mathsf{T}}, \vartheta_{p,\text{Arms}}^{\mathsf{T}}] \tag{11}$$

$$\vartheta_p = J_p \dot{q}. \tag{12}$$

5.4 Constraints Task

To prevent the robot from damaging itself or obstacles in the environment, we insert a constraint task to which we assign the highest priority. These constraints consist of joint limit avoidance, self collision avoidance and obstacle avoidance. Any action arising from the manipulation and posture tasks that would violate these constraints are filtered directly in the control loop. All three constraints rely on artificial potential fields [12] and use efficient distance computation using capsules. An example is given in Sect. 7.3.

5.5 *Hydrodynamic Feed-Forward Compensation*

O_2 experiences hydrodynamic effects when operating in an underwater environment, which is captured by the last term in (1). In order to compensate for these forces, we add a feed-forward term ($J_h^\mathsf{T} F_h$) to the controller in (2). Because this computation is part of the controller, we need to rely on models that are executable in real-time. For this purpose, we model O_2's body, upper arms, lower arms, and hands as rigid links. For each link, we compute two forces: Viscous Damping and Buoyancy. Buoyancy is computed using each link's volume and center-of-buoyancy extracted from the CAD files. For viscous damping, we use the standard model of a cylinder

$$^{B}F_D = -\frac{1}{2}\rho \begin{bmatrix} C_x \bar{r}^2 \pi \mid {}^B\dot{x} \mid {}^B\dot{x} \\ C_y 2\bar{r}\bar{L} \mid {}^B\dot{y} \mid {}^B\dot{y} \\ C_z 2\bar{r}\bar{L} \mid {}^B\dot{z} \mid {}^B\dot{z} \end{bmatrix}. \tag{13}$$

We assume a local coordinate system in each link, originating at the center and $\hat{x}$ along the cylinder axis. C_x, C_y and C_z are constant parameters, $\bar{m}$, $\bar{L}$ and $\bar{r}$ are cylinder parameters. The combined hydrodynamic force on each link is computed with

$$^{0}F_{h,i} = {}^{0}R_B \, {}^{B}F_D + {}^{0}F_B. \tag{14}$$

With the associated Jacobians we can now compute the total hydrodynamic compensation

$$\Gamma_h = (J_h^\mathsf{T} F_h) = \sum_i J_i^\mathsf{T} \, {}^{0}F_{h,i}. \tag{15}$$

5.6 *Thruster/Body Control*

Ocean One has eight thrusters, four horizontal thrusters arranged in diamond-shape to control yaw and planar translates, four vertical thrusters arranged in square-shape to control roll, pitch and vertical translation. This redundancy allows for holonomic actuation and full maneuverability in case of a thruster failure. The mapping from the eight-dimensional thruster force vector T to six-dimensional generalized force vector Γ acting on the body is given by $\Gamma = J_{\text{Thruster}}^\mathsf{T} T$, more specifically

$$\Gamma = \begin{bmatrix} J_1^\mathsf{T} n_1 & J_2^\mathsf{T} n_2 & \dots & J_8^\mathsf{T} n_8 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_8 \end{bmatrix}, \tag{16}$$

where J_i^T is the Jacobian of the ith thruster and n_i its associated unit thrust vector. The robot is controlled in terms of generalized coordinates thus the inversion of (16)

is needed. Because $J_{\text{Thruster}}^{\mathsf{T}}$ is a wide matrix, it is not immediately invertible but two more constraints are required. These constraints arise from the elimination of internal forces and moments given in

$$T_1 + T_2 + T_3 + T_4 = 0 \tag{17}$$
$$T_5 + T_7 - T_6 - T_8 = 0. \tag{18}$$

Inverting this system of equations is equivalent to solving the optimization problem

$$\text{minimize} \quad T^{\mathsf{T}}T \tag{19}$$
$$\text{s.t.} \quad \Gamma = J_{\text{Thruster}}^{\mathsf{T}}T.$$

The thrusters are limited by their force capacity $T_{\max}$. This limit is imposed by proportionally scaling back all thruster forces until none exceeds this limit.

6 Teleoperation and Haptic Interaction

The pilot and robot are directly coupled at their hands via bimanual haptic devices. The robot mimics the pilot's movements and the pilot receives force feedback that is perceived through the robot's 6D force sensors at the wrists. We refer to this mode of collaboration as *Avatar*-mode. The pilot is stationary at the console while the robot is navigating through space in a holonomic manner. To achieve this mapping, we superimpose position and rate control to compute goal position and orientation of the two hands. For this purpose we introduce an intermediary coordinate frame referred to as *Manipulation-Frame* or *Mframe*. This frame is responsible for the rate contribution, it drifts in translation and yaw in proportion to the sum and difference to the haptic devices' linear positions. The position contributions are hand translations and orientations, directly mapped from the haptic devices' positions and orientations (Fig. 5).

Fig. 5 Four planar thrusters (green T1–T4) and four vertical thrusters (magenta T5–T8) enable holonomic actuation with 2° of redundancy. Three linear and three angular forces acting on a body-fixed coordinate system (yellow) are mapped to thruster forces (T1–T8)
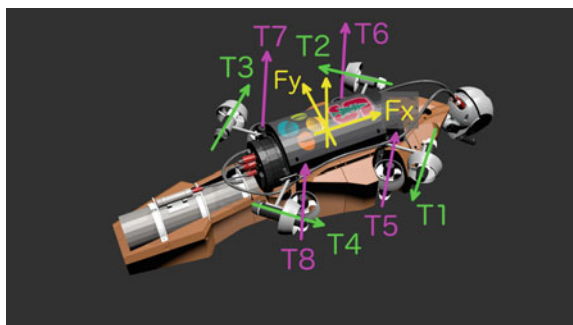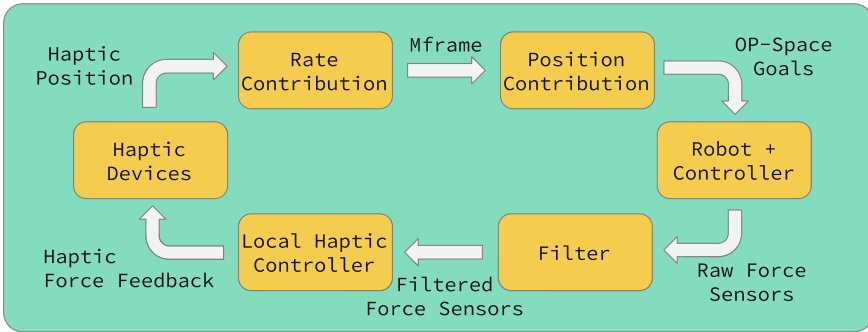
Fig. 6 Haptic teleoperation. The operational-space goals of the manipulation task are computed by a hybrid rate and position signal. The rate contribution is applied to a virtual coordinate frame (*Mframe*), and the position contribution is expressed in this frame. The controller enforces these goals. The raw force signals, measured at the wrists are filtered to remove very high frequency noise and forwarded and superimposed with a local haptic device controller, which allows the pilot to perform guided motions

These goals are then forwarded and enforced by the Manipulation task. O_2's contact forces are measured by force sensors located at the wrists. The raw signals are passed through filters in order to eliminate high frequency noise while maintaining haptic transparency. The haptic devices do not only reflect the filtered contact forces but are actively controlled. This allows the pilot to perform guided motions, which simplifies the teleoperation task for the pilot by reducing its dimensionality. For instance, certain fetching tasks only require 1 active DoF in orientation, and a docking maneuver only requires 1 linear DoF (Fig. 6).

7 Simulation

Simulation played an integral part throughout the development of O_2. Most importantly, it enabled us to develop and test O_2's software stack prior to fabrication and deployment on the physical robot. It also informed mechanical design choices and allowed us to train the pilots and practice entire missions. SAI2 is a real-time interactive simulation environment comprised of a collection of libraries that include the simulation of multi-body dynamic systems, contact and collision resolution. In addition, we utilized the Chai3D [13] libraries to facilitate haptic and visual rendering.
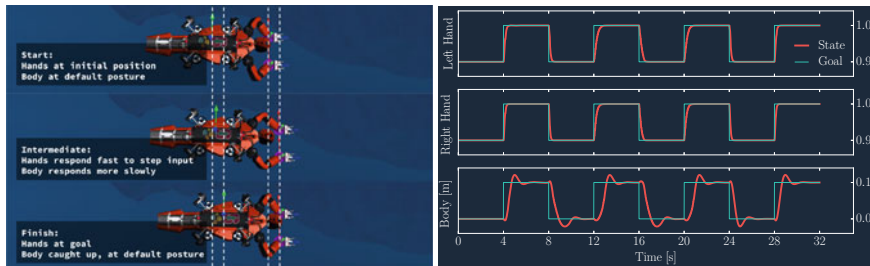
Fig. 7 A square wave goal position input is applied to the hands' operational points. We observe that the hands' dynamic responses are much faster than the body's alone. The combined macro-mini dynamics of slow body and fast arms allows for fast overall dynamics. The body is limited by its large inertia and comparatively low actuation capabilities

7.1 Step Response in Operational Space

O_2's kinematic structure can be decomposed into three parts. The body, referred to as *macro-manipulator* with 6 DoF and two arms referred to as *mini-manipulators* with 7 DoF each. This is a valid decomposition because the minis have full range in operational space and the macro has at least 1 DoF [14]. The serial combination of macro and mini offers two advantages. First, the effective inertias of the macro-mini combination is upper bounded by the effective inertias of the mini-manipulator alone. Second, the dynamic performance of the macro-mini can be made comparable to that of the mini.

This behavior is illustrated in Fig. 7. A square wave function is applied as position-goal for both hands in forward-direction, while lateral and vertical position-goals remain constant. The step-response of the body alone (bottom) shows slow dynamics with large overshoot, oscillation and long duration for convergence. This behavior is due to the body's large inertia, and comparatively weak actuation capabilities due to thruster force limitations. The step-responses of the hands' combined macro-mini dynamics display fast response with small rise-time and critical convergence. Hence, the fast, lightweight arms compensate for the slower body and overall response is fast and accurate.

7.2 Docking Maneuver with Force Control

To illustrate the advantages of whole-body coordination in force control mode, we perform a docking maneuver, where both hands apply and maintain a force normal to a given plane. The maneuver initiates at close proximity to the contact surface, where
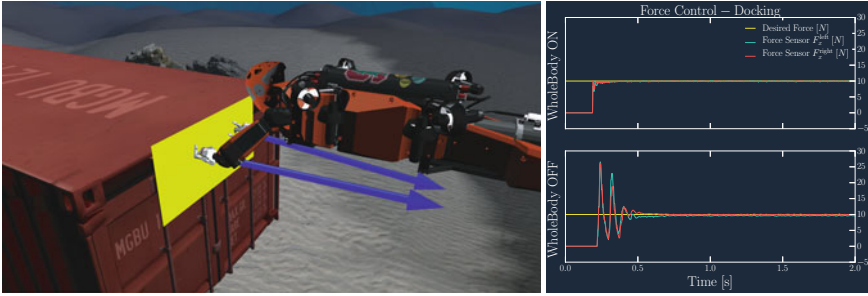
Fig. 8 We compare a docking maneuver with and without whole-body control. The hands initialize at close proximity to the red container. The collision plane is indicated by the yellow rectangle. We apply force control in forward-direction and position control in the orthogonal plane, on both hands, the desired force is 10N. The forces are rendered by the blue vectors. We observe that whole-body control leads to superior transitions (reduced spikes) and steady state behavior (faster convergence, smaller errors)

force control is activated in the forward-direction and position control is maintained in the orthogonal sub-space. Figure 8 shows the results comparing force control with and without whole-body coordination. We see that whole-body coordination leads to superior transitional and steady state behavior. The spikes during transition are greatly reduced, convergence is faster, and steady state errors are smaller.

7.3 Obstacle Avoidance

We simulate a scenario, where O_2 manipulates a container while avoiding local obstacles. To do this, we enclose O_2 in five (green) collider capsules and the obstacle in a (red) collision capsule (Fig. 9). In every servo loop, we monitor the distances between colliders and obstacles. In case a distance is smaller than the specified activation distance ρ_0, the constraint task is activated and an artificial potential field is applied to avoid the collision. In the given scenario, we program O_2 to unscrew the container's lid by specifying circle-segment trajectories at its hands. Without obstacle avoidance, O_2's body would be sweeping through the obstacle during this motion. The smallest distance between the obstacle and O_2's body is rendered by the red line segment between the blue and red spheres. Instead of colliding with the obstacle, the artificial potential field leads the body to glide over the barrel while the trajectory of the hands remains unaltered. The comparison between active and inactive obstacle avoidance is given in Fig. 9.
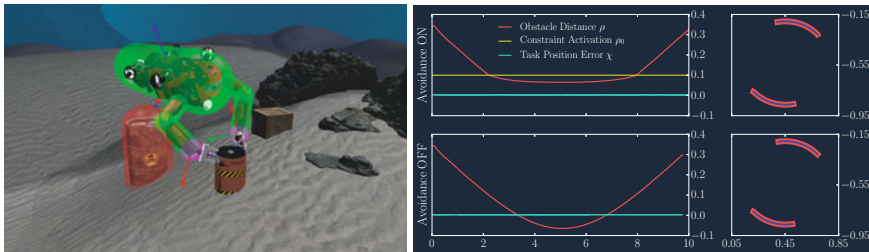
Fig. 9 Left: O_2 performs a manipulation task on the red container, while avoiding obstacles. In this scenario, O_2 unscrews the black lid of the red container. The manipulation task is programmed by applying a circular trajectory to the hands, which leads the body to rotate along. Without avoidance, this specification would lead to a collision between the tail and the rusty barrel. In order to complete the task without collision, we model the robot's body and arms collider hulls with green capsules, and the obstacle with a red capsule. The red line between the smaller red and blue sphere indicates the smallest distance between the body-collider and the obstacle. We see that as the lid is rotated, the body sweeps above the barrel while maintaining a safety margin. Right: On the left column we plot the obstacle distance (red line) and task position error (sum of the two hands' error, blue line). We see that with deactivated avoidance, the distance becomes negative, indicating collision and intersection. With activated avoidance, the constraint task triggers when $\rho < \rho_0$ (yellow line), avoiding the obstacle. The right column shows the hands' goal (thick red line) and state (slim blue line) trajectories sweep. We observe that in both cases the task is performed with the same precision

8 Deployment and Experimental Results

After O_2's hardware components were assembled, we deployed it in shallow depth at the Stanford Aquatic Center. We experimentally tuned parameters for buoyancy compensation and validated the kinematic and dynamic models as well as the sensors and communication protocols. The pool also offered the first opportunity to practice piloting the robot during navigation, grasping, and docking operations. In Fig. 10 (left) we compare the body's dynamic responses between simulated and physical robot by applying sinusoidal trajectories at 0.05 Hz and 45° amplitude to yaw and 0.3 m to depth. The responses align well with the exception of some coupling that is likely due to unmodeled hydrodynamic contributions. In Fig. 10 (right) we compare the hands' responses in operational-space position control by applying sinusoidal trajectories at 0.1 Hz to all three cartesian directions. Again, we observe good alignment between simulation and physical robot with additional coupling. The physical robot exhibits slightly decreased amplitudes, which is likely a result of under-estimated hydrodynamics and friction in the arm joints.

O_2's maiden mission took place at an archeological site in the mediterranean sea near the coast of Toulon, France. The *Lune* is a two-decked, fifty-four-gun french naval vessel of Lois XIV's that sunk in 1664 with nearly a thousand men on board to 91 m of depth, where it was discovered in 1993 by Nautile, a submarine of the French Oceanographic Institute. The mission was executed from the *Andre Malraux*, a research vessel operated by DRASSM [15]. After initial tests at 15 m depth, collaborating with a human diver, O_2 descended to a 4 h long mission to the Lune,
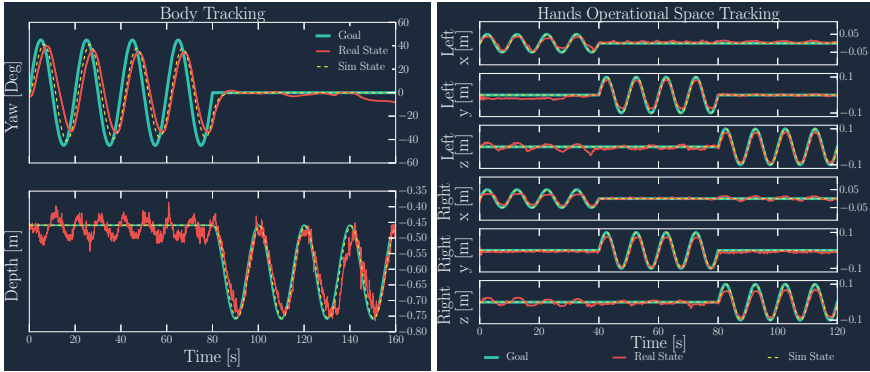
Fig. 10 Simulation Versus Real robot. Left: Dynamic Body Tracking. Sinusoidal trajectories at 0.05 Hz are consecutively tracked in yaw and depth coordinates. Simulation and real robot responses are well aligned with the exception of some coupling that is likely due to unmodeled hydrodynamic components. Right: Dynamic Operational-Space tracking. Sinusoidal trajectories at 0.1 Hz are tracked in forward (x), lateral (y), and vertical (z) directions, consecutively. Simulation and real robot are well aligned, there is more dynamic coupling and a decreased amplitude at the real robot, which is likely due to underestimated hydrodynamics and friction in the joints

Fig. 11 Ocean one deployed in Toulon, France. Left: collaboration with a diver at 15 m depth. top center/right: O_2 fetching a vase and dropping it into the collection basket. bottom center/right: researchers collecting the floated basket, Catalan vase on deck showing its acquired biofilm. (Photos courtesy of Frederic Osada and Teddy Seguin, DRASSM/stanford university

where it explored the site, fetched a vase and deposited it in a collection box that was subsequently floated to the surface.

9 Conclusion

In this paper, we focussed on O_2's control architecture. We illustrated the hierarchical implementation of whole-body control and showed how to create an immersive interface with a human pilot that enabled an avatar-like collaboration. We demonstrated the system's capabilities in simulated whole-body control, force-controlled docking maneuvers as well as a manipulation task involving autonomous obstacle avoidance. We validated the dynamic models and controller with experiments in the pool and finally established O_2's effectiveness with its deployment to its maiden mission in the mediterranean sea (Fig. 11).

References

1. Hägele, M., Nilsson, K., Pires, J.N., Bischoff, R.: Industrial Robotics in Springer Handbook of Robotics. Springer, pp. 1385–1422 (2016)
2. Groover, M.P.: Automation, Production Systems, and Computer-Integrated Manufacturing. Prentice Hall Press (2007)
3. Dudek, G., Giguere, P., Prahacs, C., Saunderson, S., Sattar, J., Torres-Mendez, L.-A., Jenkin, M., German, A., Hogue, A., Ripsman A., et al.: Aqua: an amphibious autonomous robot. Computer, **40**(1) (2007)
4. Vasilescu, I., Varshavskaya, P., Kotay, K., Rus, D.: Autonomous modular optical underwater robot (amour) design, prototype and feasibility study. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation, ICRA 2005. pp. 1603–1609. IEEE (2005)
5. Kimball, P.W., Rock, S.M.: Mapping of translating, rotating icebergs with an autonomous underwater vehicle. IEEE J. Oceanic Eng **40**(1), 196–208 (2015)
6. Schilling Robotics, LLC, 260 Cousteau Place, Davis, CA 95618, USA, Homepage. http://www.schilling.com (Accessed: December. 14, 2011). Internet (2011)
7. ECA Group, Homepage. http://www.ecagroup.com/en/solutions/arm-5e-micro (Accessed: September. 18, 2016), Internet (2016)
8. Stuart, H., Wang, S., Gardineer, B., Christensen, D.I., Aukes, D.M., Cutkosky, M.R.: A compliant underactuated hand with suction flow for underwater mobile manipulation. In: ICRA 2014, pp. 6691–6697 (2014)
9. Khatib, O., Yeh, X., Brantner, G., Soe, B., Kim, B., Ganguly, S., Stuart, H., Wang, S., Cutkosky, M., Edsinger, A., et al.: Ocean one: a robotic avatar for oceanic discovery. IEEE Robot. Autom. Mag. **23**(4), 20–29 (2016)
10. Khatib, O., Sentis, L., Park, J., Warren, J.: Whole-body dynamic behavior and control of human-like robots. Int. J. Humanoid Robot. **1**(01), 29–43 (2004)
11. Khatib, O.: A unified approach for motion and force control of robot manipulators: the operational space formulation. IEEE J. Robot. Autom. **3**(1), 43–53 (1987)
12. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. In: Autonomous Robot Vehicles 1986, pp. 396–404. Springer (1986)
13. Conti, E., Barbagli, F., Balaniuk, R., Halg, M., Lu, C., Morris, D., Sentis, L., Warren, J., Khatib, O., Salisbury, K,: The chai libraries. In: Proceedings of Eurohaptics 2003, pp. 496–500. Dublin, Ireland (2003)
14. Khatib, O.: Inertial properties in robotic manipulation: an object-level framework. Int. J. Robot. Res. **14**(1), 19–36 (1995)
15. L'Hour, M.: The french department of underwater archaeology: a brief overview. Eur. J. Archaeol. **15**(2), 275–284 (2012)