Bilateral Teleoperation of Underactuated Unmanned Aerial Vehicles: The Virtual Slave Concept

Abeje Y. Mersha, Stefano Stramigioli and Raffaella Carloni

Abstract—In this paper, we present haptic teleoperation of underactuated unmanned aerial vehicles by providing a multidimensional generalization of the virtual slave concept. The proposed control architecture is composed of high-level and low-level controllers. The high-level controller commands the vehicle to accomplish specific tasks and renders both the state and the environment of the vehicle to the operator through haptic feedback. The low-level controller interprets the command signals from the operator, regulates the dynamics of the vehicle and feeds back its state to the high-level loop. Passivity of the teleoperation loop is always ensured independently of the choice of implementation of the low-level controller and the configuration of the flying hardware by a passivity-enforcing supervisor, which associates every action of the slave with an energy expense that can only be made available from a multi-state energy tank. The effectiveness of the proposed algorithm is illustrated with simulations and experimental tests.

I. INTRODUCTION

The research field of aerial robotics, in which Unmanned Aerial Vehicles (UAVs) are used as aerial platforms for robotic activities, has opened opportunities for technological advancement in the state of the art [1]. Operating UAVs in complex application areas, such as power plant inspection by contact, sample picking and disaster management, requires human in the loop for successful task accomplishment, safety and faster responses. This is primarily due to the fact that it is difficult, if not impossible, to identify the remote environment such as space and automate all predicted tasks. On the other hand, to enable the human operator (who doesn’t need to be a pilot) focus more on the primary task the UAV is deployed for, less physical and mental load for maneuvering the platform itself is desired. This can be achieved by enhancing the awareness of the operator about both the state of the remotely operated UAV and its surrounding environment.

For situational awareness, most researches have focused on either unilateral teleoperation or bilateral teleoperation, whose feedback loop is closed through vision by either keeping the vehicle fly in line of sight of the operator or using images coming from onboard cameras [2]–[4]. While visual feedback has been employed more for teleoperation of UAVs, they have limitations emanating from the camera field of view, high dependency on weather conditions and significantly higher transmission delays compared to other control signals. Force feedback, on the other hand, has proved to be either an added value when employed in augmentation with visual feedback or intuitive and adequate enough even when used alone [5]–[8]. Obviously, its employment to maximize the tele-presence of the operator has encountered challenges of transparency and stability.

In this paper, the focus is on haptic teleoperation of underactuated UAVs. We extend the virtual slave concept introduced in [7] to multidimensional and underactuated case. We also introduce the concept of multi-state energy tank, which the action of each degree of freedom (DoF) of the slave is associated with. Moreover, we presents the first experimental validation of the concept on real flying prototypes.

We implement a shared hierarchical control strategy between the high-level teleoperation and the low-level control loops, composing the overall control structure. Due to the special implementation method used, even though the two control loops influence the performance of one another, their stability is ensured independently.

A passivity-enforcing supervisor, which allows completion of a task as long as the task falls in the category of permitted tasks and if there is energy available from a local multi-state energy tank, plays fundamental role for the stability of the high-level teleoperation loop. In the low-level controller, passivity is guaranteed by controller design choice.

The paper is organized as follows: In Sec. II, the overall hierarchical control structure, composed of the high-level teleoperation and low-level control loops, is presented. In Sec. III, validating simulations are provided. Experimental results that illustrate the task performance of the proposed algorithm are given in Sec. IV. Finally, concluding remarks are provided in Sec. V.

II. CONTROL STRUCTURE

The general control structure of the implemented teleoperation system is shown in Fig. 1. It is composed of high and low-level controller loops. The high-level controller loop includes all functional blocks responsible for transforming the operator’s input command to appropriate reference for the UAV and for rendering its state and the remote environment back to the operator through haptic feedback. The low-level controller loop, on the other hand, includes the tele-operated UAV that interacts with the environment and its controller that directly regulates its dynamics. The working principles of each loops are detailed out in the remainder of this section.
A. High-Level Controller

The main components of the high-level control loop are the master and the slave systems. In the proposed control scheme, unlike classical bilateral teleoperation, the operator remotely interacts with a virtual slave rather than the real slave. The virtual slave serves as a proxy for the real slave, as a consequence, it dynamically interacts with both the master and the real slave, and yet, provides additional features.

1) Multidimensional Virtual Slave: The virtual slave concept has been introduced primarily to overcome the pervasive dissipation inherent to flying vehicles due to the action of gravity on the real slave [7]. Consequently, the master is energetically coupled with the virtual slave rather than the real slave. As shown in Fig. 1, the complete virtual slave system, which is part of the complete slave controller, consists of the virtual vehicle, a multi-state energy tank, a passivity-enforcing supervisor and a local virtual slave controller.

a) The virtual vehicle: The virtual vehicle serves two main purposes: it generates appropriate reference commands for the real slave based on the operator’s command and dynamically maps back the state of the real vehicle to create situational awareness of its state and the environment. To serve its purposes, it mimics the motion of the real slave in a gravityless and frictionless environment.

From the teleoperation task viewpoint, it is not necessary to implement a virtual vehicle that has an equivalent dynamics over the entire $n$ DoFs as the real UAV. Given the fact that most UAVs currently available are underactuated, i.e. only $m$ actuators, teleoperation over only $m < n$ DoFs is possible. Often $m \leq 4$, since the operator can have control over translational $x$, $y$ and $z$, and rotational $\psi$ (yaw) DoFs. Hence, simulating only the dynamics of the $m$ DoFs that are directly controlled by the operator is adequate. By applying the resultant wrench acting on this virtual vehicle directly, we can treat it as a fully actuated systems. As such, we can hide the underactuatedness of the real UAV from the operator, which would result in high cognitive overload otherwise. Moreover, since the virtual slave is a part of the complete slave controller, the actuator dynamics of the real UAV doesn’t need to be included, rather their net effect as wrench acting on the $m$ DoFs of the vehicle is enough for proper coupling with the real slave.

Let $\psi^o$ and $\psi^b$ be the inertial and the body fixed coordinate frames, where $\psi^b$ is chosen to be at the center of gravity and aligned with the principal axis of the virtual vehicle. The equations of motion of the virtual vehicle in port-Hamiltonian form [9] are given by

\[
\begin{align*}
\dot{\bar{P}}_{b-v}^b &= P_{b-v}^b \partial H(\bar{P}_{b-v}^b) + \bar{W}_{b-v}^b \\
\bar{T}_{b,v} &= \partial H(\bar{P}_{b-v}^b)
\end{align*}
\]

where $\bar{P}_{b-v}^b = \bar{P}_{b-v}^b \bar{T}_{b,v}^0$ is the screw momenta of the virtual vehicle, $H(\bar{P}_{b-v}^b) = \frac{1}{2}(\bar{P}_{b-v}^b)^T (\bar{I}_{b-v})^{-1} \bar{P}_{b-v}^b$ is the Hamiltonian function, in which $\bar{I}_{b-v}$ is the inertia matrix of the virtual vehicle ($\bar{I}_{b-v}$ is diagonal due to the choice of coordinate frame $\psi^b$). $\bar{P}_{b-v}^b$ is a block-wise matrix with skew symmetric components [10]. $\bar{W}_{b-v}^b$ and $\bar{T}_{b,v}^0$ represent wrench acting on and twist of the virtual vehicle. The subscript $v$ reference to and $\bar{\cdot}$ indicates the variables of the reduced fully actuated dynamics of the virtual vehicle, which has $m$ DoFs.

The dynamic behavior of the virtual vehicle fundamentally lies on the wrench $\bar{W}_{b-v}^b$. For the virtual vehicle to faithfully convey the bidirectional message between the operator and the real slave serving as a proxy, $\bar{W}_{b-v}^b$ must depend on both the dynamic coupling of the virtual vehicle with the master, through the action of the virtual slave controller, and the viscoelastic coupling with the real slave, through the real slave controller. Mathematically,

\[
\bar{W}_{b-v}^b = \bar{W}_{b-v}^b + \bar{W}_{b-rc}^b
\]

where $\bar{W}_{b-v}^b$ and $\bar{W}_{b-rc}^b$ are wrenches due to the virtual slave controller and the viscoelastic coupling, see Fig. 1.
b) Multi-state energy-tank: The multi-state energy tank is basically used to monitor the energy of the complete system so as to ensure that no more energy is extracted from the system than is injected and originally stored, i.e. passivity condition. In this paper, we propose a multi-state energy tank whose state has a dimensions of $m$, which is equal to the DoFs of the virtual vehicle. In such a way, an elegant dynamic association between the states of the energy tank and each DoFs of the virtual vehicle, whose merits are described in the next subsection, is made.

Ideally, we can consider this energy tank as a dynamic system, in particular an ideal spring, which exchanges energy with the virtual vehicle in a specific manner. If the states of the energy tank are $\Lambda_v$, its energy content can be given by $H(\Lambda_v) = \frac{1}{2}{\Lambda_v}^T K_v \Lambda_v$, where $K_v$ is an $m \times m$ positive definite diagonal matrix. The power flow between this tank and the virtual vehicle in port-Hamiltonian form is given by

$$
\left( \begin{array}{c}
\dot{\bar{\rho}}_b^{\text{v}} \\
-\Lambda_v \\
\end{array} \right) = \left( \begin{array}{cc}
0 & \Omega_v \\
-\Omega_v^T & 0 \\
\end{array} \right) \left( \begin{array}{c}
\frac{\partial H}{\partial \bar{\rho}_b} \\
\frac{\partial H}{\partial \Omega_v} \\
\end{array} \right)
$$ (1)

where $H := H(\bar{\rho}_b^{\text{v}}, \Lambda_v) = H(\bar{\rho}_b^{\text{v}}) + H(\Lambda_v)$ is the Hamiltonian of the coupled virtual vehicle-energy tank system. $\Omega_v$ is a $m \times m$ modulated matrix which is assigned by the passivity-enforcing supervisor. Note that the interconnected system of the virtual vehicle and the energy tank is passive. Mathematically, it can be shown as

$$
\frac{dH}{dt} = \frac{\partial H}{\partial \bar{\rho}_b} \dot{\bar{\rho}}_b^{\text{v}} + \frac{\partial H}{\partial \Omega_v} \dot{\Omega}_v = \frac{\partial H}{\partial \bar{\rho}_b} \Omega_v^T + \frac{\partial H}{\partial \Omega_v} \Omega_v^T \frac{\partial H}{\partial \Omega_v} = 0
$$

Therefore, the interconnection does not generate additional energy, rather simply routes energy from the energy tank to the virtual slave or vice-versa, depending on $\Omega_v$.

c) Passivity-enforcing supervisor: The task of the passivity is to assign $\Lambda_v$, such that the desired task is accomplished. When assigning $\Omega_v$, it associates every action on the virtual vehicle with an energy cost and allows it only if there is available energy from the multi-state energy tank. For the virtual vehicle to serve its purpose, $\Omega_v$ in Eq. 1 should be influenced by both the operator’s command and the state of the real UAV.

For better task performance, we implement a more conservative passivity criterion that enforces passivity not only based on overall energy content but also on each energy state, i.e. each DoF can only consume the amount of energy allocated to it. As such, not only passivity but also desired controllability is achieved (See Sec.III for illustration of the merit of the multidimensional tank). Note that by the particular choice of $\psi^b$, $\Omega_v$ also becomes a diagonal matrix. Beyond ease of computation, the fact that $\bar{\rho}_b^{\text{v}}$ is diagonal enables an elegant decoupled one-to-one association of the action on each DoF to each energy state. The diagonal elements $\Omega_v(i, i)$ are assigned as

$$
\Omega_v(i, i) = \begin{cases} 
0, & \text{if } \Lambda_v(i) < \gamma \text{ and } W_b^{b,v}(i)T_{b,v}^{b,b} > 0 \\
\frac{1}{\Lambda_v(i)}W_b^{b,v}(i), & \text{otherwise}
\end{cases}
$$ (2)

In Eq. 1 the initial states of the tank $\Lambda_v(0)$ can be chosen freely, as deemed necessary for the task. However, one can associate it to the maximum amount of energy that can be consumed from the tank to achieve the maximum allowable twist $T_{b,v}^{b,b,max}$, so that the virtual vehicle will not move beyond, i.e.,

$$
T_{b,v}^{b,b}(i) = \sqrt{\left\{ \Lambda_v^2(i, i) - \gamma^2 \right\} K_v(i, i)}.
$$

d) The virtual slave controller: The task of this controller is to effectively track the reference twist commanded by the operator. Any kind of velocity controller suitable for the task can be employed without compromising its passivity, for it is guaranteed by the passivity-enforcing supervisor. We propose, however, to employ a simple proportional-integral controller on the twist of the virtual slave for better disturbance rejection.

2) The Master: The main component of the master system is a haptic interface by which the operator inputs the command and receives a force feedback. The two most important aspects with regard to this system are the command mapping and the passivity of the system along with the choice of force feedback, which are discussed in the sequel.

a) Mapping: Unlike in most classical teleoperation tasks, the master is neither an exact nor a scaled replica of the slave. Due to the incompatibility between the bounded and the unbounded workspaces of the master and the slave respectively, a direct state mapping is not feasible for most tasks. Hence, we adopt a rate control strategy, i.e. mapping the position of the master to the desired velocity of the slave. However, since common haptic interfaces have compatible rotational workspace with the slave, it is possible to have direct state mapping for the rotational dynamics.

In our algorithm, the pose of the tip of the haptic interface, denoted by a 6 dimensional vector $q_{tip}^0(t)$, is mapped to the desired twist $T_{b,v}^{b,b}$ of the slave, i.e.,

$$
q_{tip}^0(t) \rightarrow \phi(Z, \alpha, A_H^b) \rightarrow T_{b,v}^{b,b}
$$

where $\phi$ is the mapping function that depends on the transformation matrices $Z$ and $A_H^b$, and the desired scaling factor $\alpha$. $A_H^b$ is an adjoint coordinate transformation from $\psi^b$ to $\phi^b$. The kinematic relation between $q_{tip}^0(t)$ and $T_{tip}^0$ is

$$
q_{tip}^0(t) = ZT_{tip}^0[10]
$$

Note that since UAVs are in general underactuated, even if $q_m$ is 6 dimensional, only the possible and desired pose commands of the tip of the master are mapped to the desired twist of the virtual vehicle.
b) Master dynamics: Common serial haptic interfaces with \( l \) DoFs can be modeled as

\[
\begin{pmatrix}
\dot{r}_m \\
\dot{P}_m \\
\end{pmatrix} = \begin{pmatrix} 0 & I \\ -I & -R \end{pmatrix} \begin{pmatrix}
\frac{\partial H(r_m, P_m)}{\partial r_m} \\
\frac{\partial H(r_m, P_m)}{\partial P_m} \\
\end{pmatrix} + \begin{pmatrix} 0 \\ I \end{pmatrix} \tau
\]

(3)

where \( r_m \) and \( P_m \) represent the \( l \) dimensional generalized joint level configuration and momentum of the master, respectively. \( R \) is an \( l \times l \) positive definite diagonal matrix representing viscous damping and \( \tau \) is the generalized force acting on the joints. The Hamiltonian function of the system is given by the sum of kinetic and potential energies as

\[ H(r_m, P_m) = \frac{1}{2} (P_m) ^T M^{-1} (r_m) P_m + U(r) \]

in which \( M(r_m) \) is an \( l \times l \) positive definite inertia matrix. Hence, the high-level teleoperation loop is passive at all operating conditions.

In most teleoperation tasks however, it is common and intuitive to map the state (position, velocity) of the haptic interface as a command for the slave and display a force feedback at this tip. Hence, it is required to describe the dynamics of the master system in cartesian coordinates. By first transforming the port variables \((\dot{r}_m, \tau)\) to new variables \((\dot{q}^0_{tip}, \dot{W}^0_{tip})\) using the geometric jacobian, followed by few mathematical manipulations that also involves treating the gravitational force as external wrench, we can describe the tip dynamics of the haptic interface by

\[
\begin{pmatrix}
\dot{\tilde{q}}^0_{tip} \\
\dot{\tilde{W}}^0_{tip} \\
\end{pmatrix} = -\tilde{R} \frac{\partial H(\tilde{P}^0_{tip})}{\partial \tilde{P}^0_{tip}} + \tilde{W}^0_{tip}
\]

(4)

where \( H(\tilde{P}^0_{tip}) = \frac{1}{2} (\tilde{P}^0_{tip}) ^T \tilde{I} (\tilde{q}^0_{tip})) ^{-1} \tilde{P}^0_{tip} \) is the Hamiltonian in the new coordinates, in which \( \tilde{P}^0_{tip} \) and \( \tilde{q}^0_{tip} \) represent the momentum and the pose of the tip, extracted from \( H(\tilde{P}^0_{tip}) \) \( \tilde{R} \) are the inertial and the viscous friction matrices in the new coordinates. \( W^0_{tip} \) is the resultant wrench applied by gravity \( W^0_{tip-g} \), the operator \( W^0_{tip-h} \) and the master controller \( W^0_{tip-mc} \), i.e.,

\[ W^0_{tip} = W^0_{tip-g} + W^0_{tip-h} + W^0_{tip-mc} \]

c) Master controller: Assuming that the human operator is passive, the passivity of the master system boils down to the passivity of its local controller since the haptic interface is an intrinsically passive mechanical system. Hence, the design of this controller has to both guarantee passivity of the master system and generate force feedback displayed to the operator.

Assuming that an active controller separately carries out gravity compensation, the master controller \( W^0_{tip-mc} \) is the force feedback displayed to the operator. We propose a spring-damper like force feedback, which is mainly proportional to the deviation of the slave from the commanded reference. By grasping the haptic device at the desired configuration, a force feedback is felt by the operator whose magnitude and direction depend on how large and in which direction that the twist of the vehicle \( \tau^b_v \) deviates from the desired twist \( \tau^b \).

To ensure the passivity of the master controller, we endow it with a local energy tank that collects the energy supplied by the operator and virtually dissipated by the master’s controller action. The energy tank, whose content is denoted by \( E(t) \), should be pre-filled according to the desired force displaying capability and bounded from above by \( E_{max} \) to avoid unlimited storage of energy that may result in unwanted transitional behavior and even loss of passivity.

Consider the proposed controller with its energy described by the Hamiltonian function \( H(q^0_{tip}) = \frac{1}{2} (q^0_{tip}) ^T K_{pm} (q^0_{tip} - q_v(t))^2 \)

where \( q_v(t) \) is a \( 6 \) dimensional vector representing the pose corresponding to the twist of the virtual slave, and \( K_{pm} \) is a positive definite matrix.

With the above controller, the force \( W^0_{tip-mc} \) displayed to the operator is given by

\[
\begin{cases}
0, \\
-Z^T K_{pm} (q_m - q_v) - K_{dm} T^0_{tip}, \text{ otherwise}
\end{cases}
\]

(5)

where \( K_{dm} \) is a positive definite matrix representing the virtual damping, which is added to reinforces the existing intrinsic damping action of the haptic device. This helps to avoid undesired oscillatory behavior that may result during light grasp and high force feedback. \( E(t_0) \) is the initial energy stored in the master tank.

Unlike other energy based bilateral teleoperation, it is not required to exchange energy between the master and the slave system through the unreliable communication channel. This is mainly attributed to the mapping strategy, where for instance, a constant positon of the haptic interface that corresponds to zero power injection from the operator, commands a velocity, where energy is perhaps required. The pre-filled local multi-state energy tank of the virtual slave discussed in Sec. II-A.1 serves as a virtual energy that is transported from the master.

Fig. 2 shows the control structure shown in Fig.1, where some components are grouped together for intuitive explanation of the passivity analysis. To claim that the high-level teleoperation loop is passive, it suffices to shown that the big circular block in Fig. 2 is passive since the other components are either intrinsically passive or assumed to be passive in the desired frequency range (the human operator). This approach relaxes the conservativeness of the passivity criteria, which otherwise should be applied to each component of the block.

This block has two multidimensional power ports, by which it interacts with the virtual vehicle and the haptic interface. As detailed out in Sec. II-A.1 and in this section, passivity is ensured not to be violated at both ports. Hence, the high-level teleoperation loop is passive at all operating conditions.
TABLE I
PARAMETERS (IN SI UNITS) USED IN SIMULATIONS. THEY CORRESPOND TO YAW, X, Y AND Z COORDINATES. $d := \text{diag}$

<table>
<thead>
<tr>
<th>Virtual Slave</th>
<th>Master</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T^v_{\text{dc}} = d(0.3, 15, 15, 15) \times 10^{-2}$</td>
<td>$K_{\text{p,env}} = \text{diag}(1, 5, 5, 5)$</td>
</tr>
<tr>
<td>$K_{\text{p,v}} = d(2, 10, 10, 10)$</td>
<td>$K_{\text{d,v}} = d(1, 1, 1, 1)$</td>
</tr>
<tr>
<td>$R = d(0.01, 5, 5, 5) \times 10^{-2}$</td>
<td>$E_{\text{max}} = (0, 2, 2, 2)$</td>
</tr>
</tbody>
</table>

B. Low-level controller

The low-level controller is the controller of the real slave. Its main goal is to effectively track the desired states of the virtual vehicle by generating appropriate actuator inputs. The controller can be employed for tracking purpose as in [10]-[12] or obstacle avoidance as in [6], [8], or both. Since our teleoperation algorithm works in a plug and play fashion with any type of lower level controller deemed necessary for the task at hand, designing one is not in the scope of this work. However, we describe its role in our algorithm.

The low-level controller plays the role of coupling the virtual and the real vehicle. It takes the states of the virtual vehicle as a reference to the real slave. It feeds back the wrench $\dot{W}_{b-r,b}^v$, i.e., the negative of the resultant wrench applied on the real vehicle to the high-level teleoperation circuit through the virtual slave to provide information on the state and the environment of the real UAV. Stronger viscoelastic coupling increases the fidelity of the information transmitted to the operator. Note that it is based on this resultant wrench, which is responsible for the viscoelastic coupling between the real and virtual slave, that actual control inputs and then actuator allocations are carried out for control of the underactuated real UAV.

As a consequence of such coupling between the master and the real slave through the virtual slave, the low-level controller influences only the performance of the teleoperation loop. Thanks to the passivity-enforcing supervisor employed its stability is however, independent of it. Complementarily, the stability of the low-level control loop is independent of the high-level control loop.

III. SIMULATION RESULTS

In this section, two representative simulation results that show the effectiveness of the proposed algorithm and the added value of using the multi-state energy tank are presented. In both simulations, a shared hierarchical control structure, in which the operator tele-controls the translational motions of the UAV while the yaw is controlled autonomously, is implemented. The ducted fan miniature UAV [11] and the low-level controller proposed in [10] are used as real slave and lower level controller. A 2s time delay in both the forward and backward communications are included. Besides, wind gust of 15N and -25N are applied at the center of gravity of the UAV along its y and z axes between 30s-40s and 25s-40s, respectively. The mapping scale $\alpha = 8$. Parameters used during the simulations are given in Table I.

In the first simulation, a single state energy tank is implemented in the virtual slave. Fig. 3b shows that the translational velocities of the virtual and real slave follow the command imposed by the operator, which is depicted in Fig. 3a, until all the energy in the tank is used up. When a wind gust acting on the UAV deviates its velocity, it also deviates the velocity of the virtual vehicle from the desired reference due to the viscoelastic coupling. This deviation of the slave due to disturbance and communication time delay are reflected back to the operator as a force feedback, as shown in Fig. 3d.

When the disturbance is strong enough to deviate it beyond the desired maximum velocity, the virtual vehicle also deviates so long as there is energy present in the local tank. Hence, the velocity in one axis may go beyond the desired maximum and constrain the motion in the other axes.
by consuming the available energy. This instance is shown in Fig. 3b at around 30s-45s, in which the motion along the y-axis constrains the motion along the z-axis for some time. This happens mainly because of the type of passivity enforcing supervisor implemented, i.e., passivity based on the overall energy. However, when implementing the proposed more conservative passivity criterion for similar conditions in the second simulation, we obtain a better result, as shown in Fig. 3c. Even in the existence of strong disturbance, the controller is able to track the desired trajectory commanded by the operator without consuming other DoFs energy, and hence without acquiring a virtual slave velocity more than the specified maximum, i.e., 2m/s in these simulations.

The simulation results presented above show that the one to one association between the DoFs of the virtual vehicle and the states of the energy tank enable the operator to have a better control of the task on each DoF. Moreover, they validate the passivity, thereby stability, and performance of the teleoperation control algorithm even in the presence of practically large communication delays.

IV. EXPERIMENTAL RESULTS

The algorithm discussed and validated through simulation was physically realized on three aerial platforms. The first two prototypes, namely, the coaxial rotorcraft and the ducted fan underactuated flying vehicles were developed by ETH-Zurich and University of Bologna in the European project AIRobots [1]. Fig. 4 shows the two prototypes. The third platform is the AscTech Pelican quadrotor. During the first two experiments the Phantom Omni haptic device from SensAble Technologies and the tracking controller algorithm proposed in [11] were used as haptic interface and lower level controller, respectively. In the last experiment the Force Dimension Omega.6 haptic interface and the lower level controller proposed in [13] were used.

In all experiments, a shared hierarchical control strategy between the autonomous and the teleoperation control was carried out. In the first, the operator controlled the z-axis and the autonomous controller controlled the rest DoFs, whereas in the third experiment the operator controlled all the translational DoFs and the yaw was controlled by the autonomous controller. Moreover, a mapping scale of \( \alpha = 1 \) and \( \alpha = 4 \) were used between the master position and the slave velocity the first two experiments and the third experiment, respectively. Note that the lower-level controller employed in the first two experiments were not optimized. Nonetheless, the main focus of this part of our work is to experimentally verify the proposed teleoperation algorithm, not obtaining the best tracking performance.

In the first experiment, our algorithm was implemented on the ducted fan prototype and we used a virtual haptic interface emulating the Phantom Omni. Fig. 5a shows the position of the master device and the velocities of both the real and virtual vehicles from this experiment. It can be seen that the desired virtual vehicle velocity derived from the position of the master device is tracked with a certain lag due to the delay introduced in the communication and in the actuator dynamics. At the start of the teleoperation the tracking performance is low because of the difference between the initial velocities of the real and virtual vehicles, i.e., the real vehicle picks up its previous nonzero velocity acquired during the autonomous flight, whereas the virtual vehicle starts from zero. However, during the course of the teleoperation, the tracking performance improves as desired. Fig. 5b shows the force feedback displayed to the operator during the teleoperation, which indicates the magnitude and direction of deviation of the slave from the operator’s command.

In the second experiment, the algorithm was implemented on the coaxial rotorcraft and the Phantom Omni haptic device was integrated in the teleoperation loop. The other conditions were the same as in the previous experiment. Compared to the previous experiment, the result of this experiment shown in Fig. 6 has some differences. The communication delay implemented was large and, as a result, the desired command is accomplished after a longer delay. Moreover, the
presence of the real operator in the loop has also introduced additional dynamics. As long as the operator is able to compensate for the force feedback and keeps on imposing the desired trajectory on the master side, which is not completely achievable, it is possible to get a better result. However, due to the reaction of the user to the force feedback, deviation in the position of the master device is observed, which results in a fast changing reference trajectories. This in turn results in lower tracking performance compared to the previous experiment. Nevertheless, as can be seen from Fig. 6, it is possible to haptically teleoperate the UAV in an intuitive manner, even in the presence of practically larger time delay, and yet, preserving passivity.

The third experiment shows the realization of the teleoperation algorithm in multiple DoFs. Fig.7 shows the 3D positions of the master, which is mapped to velocity reference for the virtual slave by a scaling factor $\alpha = 4$, and the velocities of the virtual and real slaves. In this experiment, the delay in the communication channel was even larger than the previous experiments. As can be seen in these plots, the operator is still able to maneuver the slave device in a wider workspace with a reasonable tracking performance using the limited storks of the Omega.6 haptic device. The tracking performance along the $z$ axis is better that the other axes due to the well tuned controller parameter along $z$. As a consequence, the deviation of the virtual slave from the operator’s command in this axis is less, which is manifested by the relatively lower magnitude of the force feedback displayed to the operator, as shown in Fig. 8.

![Fig. 7. 3D positions of the master device and velocities of the virtual and real slaves](image_url)

![Fig. 8. Force Feedbacks.](image_url)

The experimental results presented in this section demonstrates the effectiveness of the proposed algorithm. They show that the virtual slave has reasonably served as a dynamic map between the master and the real slave by generating a desired reference for the real slave based on the user’s command and the real slave’s reactions due to the user’s and environmental stimulus through the viscoelastic coupling. They also verified that force feedbacks displayed to the operator realize a certain level of transparency to the state of the slave.

V. CONCLUSIONS

In this paper, we presented a multidimensional generalization of the virtual slave concept for haptic teleoperation of underactuated UAVs. In the theoretical formulation of the algorithm, it has been shown how to map the limited workspace of the haptic interface to the unlimited workspace of the slave UAV and accomplish tasks with a good performance using the proposed multi-state energy tank. Simulation results validated the proposed haptic teleoperation algorithm. Moreover, supplemental experimental tests on three flying platforms and different trajectory tracking controllers showed that the algorithm is generic and works with any type of lower level controller and hardware configuration of the UAV.

REFERENCES

[1] Innovative Aerial Service Robots for Remote Inspection by Contact (AIRobots), http://www.airobots.eu