Aerial manipulator interaction with the environment

Santos M. Orozco-Soto, Eugenio Cuniato, Jonathan Cacace, Mario Selvaggio, Fabio Ruggiero, Vincenzo Lippiello, Bruno Siciliano

Abstract

This chapter investigates the problem of an aerial manipulator interacting with the environment. The chapter is split into two parts. The former considers an aerial device with tilting propellers that, thanks to a super-twisting slide mode controller, can control the interaction force for inspection task purposes. The latter proposes a hardware-in-the-loop simulator for human cooperation and environmental interaction with an aerial manipulator. This part includes the mathematical background and theoretical derivation with insights on the relative stability proofs. Simulations in a highly realistic environment endowed with a physics engine and real experiments validate both the proposed approaches.

Introduction

As the name suggests, aerial manipulation means the possibility to bestow manipulation capabilities to unmanned aerial vehicles (UAVs). Aerial manipulators are nowadays more and more employed in maintenance and construction tasks [1]. Their strength resides in the agility these systems offer, as they are not limited by distance and height as much as human workers usually are, but also in accurate and powerful manipulation capabilities. Other possible actions are given by grasping, transporting, positioning, assembling, and disassembling mechanical parts, instruments, measurement units, and objects.

Aerial manipulators are compound by a UAV with either a gripper (or a tool) or a several degree-of-freedom (DoF) robotic manipulator(s) on board. In the former case, also known as flying hand (FH), the object cannot be moved independently from the UAV. The latter case provides versatility and dexterousness instead. A gripper can be directly attached to the UAV in the FH case. In this circumstance, mechanical design plays a crucial role. Another solution is to use passive cables or tether mechanisms. Nevertheless, the elasticity of these structures poses several problems in modelling and simulating these devices through reliable software,

complicating the controller design. Through FHs, more than pick-and-place operations are not possible. For this reason, a UAV equipped with a robotic arm with several DOFs increases dexterity in the manipulation task. If the presence of the carried object in the FH case creates coupling effects in the dynamic model of the system, the robotic arm in an aerial manipulator provides even more issues since its dynamics depend on the system's current configuration state. Basically, for this latter case, two approaches tackle the control problem. The first is a centralised approach, where the UAV and the arm are seen as unique entities to be modelled and controlled. The second is a decentralised approach, where the UAV and the arm are seen as two separate and independent systems interacting each other. Recent states of the art regarding aerial manipulation can be found in [2], [3].

The UAVs usually employed in aerial manipulators are multi-rotor vehicles. These have conventional designs with parallel-axes rotors having the property of being underactuated and strongly coupled systems. Designs with parallel-axes rotors are divided concerning the propulsion configuration. The most common designs incorporate single propulsion units (quadcopter, hexacopter, octocopter) or coaxial propulsion units [4]. Non-parallel-axes rotors designs with tilted or tilting rotors can instead overcome the inherent underactuated property of parallel-axes rotors designs [4], but they anyway represent a challenging problem from the control engineering perspective. Another consideration for designing the control system of an aerial manipulator is the possible interaction with the environment, even including other UAVs, aerial manipulators, or humans. Indeed, aerial manipulators can help humans with their daily activities, especially in tasks where the human judgement or physical intervention is still necessary. As the human workspace becomes more and more crowded with these aerial robots, it becomes of paramount importance to understand how the interaction between the two is established, especially in terms of safety for the human operators.

Therefore, this chapter investigates the control design for an aerial manipulator interacting with the surroundings. In the first part, the aerial manipulator should deliver a sustained force to the environment (*i.e.*, a wall). A decentralized control approach is presented for sustained force delivery tasks, in which the UAV is independently controlled from the arm. Since such a robotic arm is equipped with position-source actuators, the main scope of the chapter's first part is dealing with the omnidirectional flight control of the UAV and exploiting this property to deliver lateral force upon a vertical flat surface. A second-order sliding mode control approach is implemented for trajectory tracking of the omnidirectional UAV, which switches to a "transparent" force regulator since it is provided with tilting propellers. In the chapter's second part, the aerial manipulator deals with the safe interaction with humans. In this regard, the use of a hardware-in-the-loop simulator for human cooperation has different advantages: (i) it may play the role

of a training interface work workers since it allows for physical interaction without the intrinsic danger of rotary-wing platforms; (ii) it allows the introduction of software safety layers, like a predetermined bounding box for the manipulator; (iii) it lends itself to developing and testing both autonomous and human-aerial manipulator interaction control strategies.

Both chapter's parts rely on using a simulation environment endowed with a physics engine to assess the control effectiveness, which represents a step beyond the traditional numerical simulation since the results are closer to the actual behaviour of the real platform.

The work presented in this chapter is split into two parts: the first one details the design and implementation of a model-free robust control technique for an aerial manipulator compound by a tilting (non-parallel-axes) quadrotor and a 6-degree-of-freedom (DOF) manipulator on board. The experiments showing the performance of the proposed controller were carried out within the Gazebo simulator, a widely employed robot simulator with physics engines, and consisted of flight trajectory tracking and horizontal force delivery. The second part, on the other hand, is related to developing a hardware-in-the-loop simulation system for interaction tasks. Such a setup is compounded by a simulated standard parallel-axes quadrotor with a 6-DoF arm on board, a hardware interface to enable the force-based interaction task and a middleware. Two experiments show the success of the proposed system, namely a human-drone interaction task and the installation of a bird diverter into an electrical wire.

Preliminaries and notation

In this section, general notation is introduced. Local symbols for each part are left in the related sections.

Let $\mathbf{x}(t) = \begin{bmatrix} x_1(t) & \cdots & x_n(t) \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^n$ denote a vector, the following operator is defined as in [5]

$$\lfloor \mathbf{x}(t) \rceil^q = \begin{bmatrix} |x_1(t)|^q \mathsf{sign}(x_1(t)) \\ \vdots \\ |x_n(t)|^q \mathsf{sign}(x_n(t)) \end{bmatrix}, \tag{1}$$

where $q \in \mathbb{R}$ denotes a constant equal for each component $x_i \in \mathbb{R}$, with $i = 1, \ldots, n$, of the vector $\mathbf{x}$. Its time derivative is given by

$$\frac{d}{dt} \lfloor \mathbf{x}(t) \rceil^q = q \ \mathbf{J}\left(\lfloor \mathbf{x}(t) \rceil^{q-1} \right) \dot{\mathbf{x}} = q \ \mathrm{diag}\left(|x_i(t)|^{q-1} \right) \dot{\mathbf{x}}, \tag{2}$$

where $\mathbf{J}\left(\lfloor x(t) \rceil^{q-1} \right) \in \mathbb{R}^{n \times n}$ denotes the Jacobian of $\lfloor x(t) \rceil^q$. Notice that (1) and (2) are continuous functions as long as $q \neq 0$.

The identity matrix of proper dimensions is represented by the symbol $\mathbf{I}_n \in \mathbb{R}^{n\times n}$. The zero matrix of proper dimensions is given by $0_{n\times m} \in \mathbb{R}^{n\times m}$. The zero vector is denoted by $0_n \in \mathbb{R}^n$. The vector $\mathbf{e}_i \in \mathbb{R}^3$ denotes the ith standard basis vector. The matrix denoting the rotation of an angle $x \in \mathbb{R}$ around each standard basis vector is $\mathbf{R}_{e_i}(x) \in SO(3)$, with $i = 1, 2, 3$. The symbolism $\mathbf{S}(\mathbf{x}) \in \mathbb{R}^{3\times 3}$ denotes the skew-symmetric operator of a generic vector $\mathbf{x} \in \mathbb{R}^3$. For space reasons, the functions $\cos(x)$ and $\sin(x)$, with $x \in \mathbb{R}$ a generic angle, will be shortened with c_x and s_x, respectively. Cartesian norms are employed in this chapter and they are denoted with the symbol $\|\cdot\|$.

The position and orientation of the UAV are defined through the frame Σ_b attached to the body center of mass (CoM), whereas the inertial frame is represented by Σ_w. The position of Σ_b in Σ_w and its attitude are denoted by $\mathbf{p}_b = \begin{bmatrix} x, y, z \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^3$ and $\mathbf{R}_b \in SO(3)$, respectively. The attitude can also be expressed in a minimum way through the roll-pitch-yaw Euler angles family, $\mathbf{o}_b = \begin{bmatrix} \phi, \theta, \psi \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^3$. The linear and angular velocity of Σ_b in Σ_w are denoted by $\dot{\mathbf{p}}_b \in \mathbb{R}^3$ and $\boldsymbol{\omega}_b \in \mathbb{R}^3$, respectively. The pose of the UAV can be stacked in the vector $\boldsymbol{\xi} = \begin{bmatrix} \mathbf{p}_b^{\mathrm{T}}, \mathbf{o}_b^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^6$. The position, velocity, and acceleration of the on-board manipulator joints are given by $\mathbf{q} \in \mathbb{R}^{n_j}$, $\dot{\mathbf{q}} \in \mathbb{R}^{n_j}$, and $\ddot{\mathbf{q}} \in \mathbb{R}^{n_j}$, respectively, with $n_j > 0$ the number of joints. The manipulator joints are controlled through the torque input vector $\boldsymbol{\tau}_q \in \mathbb{R}^{n_j}$. As for the UAV, a distinction is made. In the first part of this chapter, a quadrotor with tilting propellers is considered. Therefore, the control input vector is $\mathbf{u}_t^b = \begin{bmatrix} u_x, u_y, u_z, u_\phi, u_\theta, u_\psi \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^6$, expressed in Σ_b. In the second part of this chapter, the quadrotor cannot tilt its propeller, resulting in the classic parallel-axes quadrotor whose control input vector is $\mathbf{u}_f^b = \begin{bmatrix} u_T, \boldsymbol{\tau}^{b^{\mathrm{T}}} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^4$, with $u_T > 0$ the total thrust and $\boldsymbol{\tau}^b \in \mathbb{R}^3$ the control torques acting around the Σ_b axes and expressed in Σ_b.

Additionally, TABLE 1 contains the intrinsic parameters of the UAV.

Part I - Super-twisting sliding mode controller for an omnidirectional aerial manipulator

Related work and contribution

Sliding mode control (SMC) is a powerful tool for controlling disturbed uncertain systems, but also it is well known for its chattering effect as the main drawback, at least for mechanical systems. On the other hand, several techniques deal with this effect, such as replacing the discontinuous function for a continuous one or the use of higher-order sliding mode techniques [6]. In this context, the super-twisting sliding mode control (STSMC), a second-order SMC type, offers a suitable alternative with the following advantages: (i) the chattering effect is significantly

TABLE 1: Symbols used to represent the parameters of the UAV.

Symbol	Meaning
$m > 0$	Total mass of the UAV
$g > 0$	Gravity acceleration
$l > 0$	Length of the boom from the center of the airframe to the rotor
$k_f > 0$	Thrust coefficient
$k_m > 0$	Drag coefficient
$\sigma = \dfrac{k_m}{k_f} > 0$	Thrust-drag ratio coefficient
ς_i	Orientation of the ith propeller with respect to the airframe
ϑ_i	Spin sense of the ith propeller, 1 if clockwise and -1 if counterclockwise. For the specific case it $\vartheta_1 = -1$, $\vartheta_2 = 1$, $\vartheta_3 = -1$, and $\vartheta_4 = 1$
$\varpi_i \in \mathbb{R}$	Spin velocity of the ith propeller
$\alpha_i \in \mathbb{R}$	Tilting angle of the ith propeller with respect to the airframe horizontal plane

reduced by replacing the discontinuous control for a continuous one; (ii) the chattering is hidden behind an integral action; (iii) such an integral action increases the robustness. These advantages have been recently exploited by researchers for controlling quadrotor UAVs with both numerical simulation and successful experimental results [7], [8]. For the case of omnidirectional multi-rotor, second-order SMC has also been applied in simulation for tilting quadrotors and tilted hexarotors [9], [10], [11], nonetheless, authors do not take into account the allocation, which is critical for mapping the computed six-dimensional control wrench to the thrust that the tilting rotors must supply. Therefore, a more reliable result was presented in [12], where a model-based integral SMC is used.

Regarding aerial manipulators, high-order SMCs have also been implemented. Recent researches have successfully tested, in numerical simulation, both terminal sliding mode control and STSMC for non-tilting quadrotor-based aerial manipulators [13], [14], [15], which are also the closest results to those presented in this work.

The contributions presented in this part of the chapter with respect to the current state of the art are the following: (i) this chapter deals with the control of an omnidirectional quadrotor with actively-tilting propellers and a 6-DoF manipulator on-board, differently from traditional quadrotor-based aerial manipulators; (ii) the proposed controller is an STSMC for full pose tracking, taking into account the allocation matrix, that is missing in the related literature; (iii) the presented controller is extended to a hybrid pose/force regulation for sustained lateral

force delivery, which also a novelty to the latest results; (iv) the delivered force is controlled and sustained for at least 60 seconds, which is a long-time exerted force up the authors' best knowledge.

Omnidirectional manipulator dynamics

The generalised aerial manipulator dynamics can be found in [16]. However, in this part of the chapter, we consider the manipulator actuated by servomotors and controlled in position. Therefore, we consider all the disturbances from the manipulator to the UAV lumped in the gravity, Coriolis, and centrifugal terms. Thus, for the design of the controller, we start from the following mathematical model

$$\ddot{\boldsymbol{\xi}} = \begin{bmatrix} m\mathbf{I}_3 & 0_{3\times3} \\ 0_{3\times3} & \mathbf{M}_b(\mathbf{o}_b) \end{bmatrix}^{-1} \left(\bar{\mathbf{R}}_b \mathbf{u}_t^b - \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & \mathbf{C}(\mathbf{o}_b, \dot{\mathbf{o}}_b) \end{bmatrix} \dot{\boldsymbol{\xi}} - \boldsymbol{\varphi}(g, \mathbf{q}, \dot{\mathbf{q}}) \right), \tag{3}$$

with $\mathbf{M}_b(\mathbf{o}_b) \in \mathbb{R}^{3\times3}$ the symmetric and positive definite (provided that $\theta \neq \pm\pi/2$ [17]) inertia matrix of the UAV's angular part expressed as function of the roll-pitch-yaw Euler angles, $\bar{\mathbf{R}}_b = \text{blockdiag}(\mathbf{R}_b, \mathbf{R}_b) \in \mathbb{R}^{6\times6}$, $\mathbf{C}(\mathbf{o}_b, \dot{\mathbf{o}}_b)$ the Coriolis matrix of the UAV system, and $\boldsymbol{\varphi}(g, \mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^6$ the vector containing the gravity and the lumped disturbance effects from the manipulator. Breaking down the control inputs mapping yields

$$\bar{\mathbf{R}}_b \mathbf{u}_t^b = \begin{bmatrix} u_x(c_\psi c_\theta) + u_y(c_\psi s_\theta s_\phi - s_\psi c_\phi) + u_z(c_\psi s_\theta c_\phi + s_\psi s_\phi) \\ u_x(s_\psi c_\theta) + u_y(s_\psi s_\theta s_\phi + c_\psi c_\phi) + u_z(s_\psi s_\theta c_\phi - c_\psi s_\phi) \\ -u_x(s_\theta) + u_y(c_\theta s_\phi) + u_z(c_\theta c_\phi) \\ u_\phi \\ u_\theta \\ u_\psi \end{bmatrix}. \tag{4}$$

The control crossed-terms can be considered as matched disturbances affecting the same channels of the non-crossed control inputs as expressed below

$$\begin{aligned} \bar{\mathbf{R}}_b \mathbf{u}_t^b &= \begin{bmatrix} u_x(c_\psi c_\theta) \\ u_y(s_\psi s_\theta s_\phi + c_\psi c_\phi) \\ u_z(c_\theta c_\phi) \\ u_\phi \\ u_\theta \\ u_\psi \end{bmatrix} + \begin{bmatrix} u_y(c_\psi s_\theta s_\phi - s_\psi c_\phi) + u_z(c_\psi s_\theta c_\phi + s_\psi s_\phi) \\ u_x(s_\psi c_\theta) + u_z(s_\psi s_\theta c_\phi - c_\psi s_\phi) \\ -u_x(s_\theta) + u_y(c_\theta s_\phi) \\ 0 \\ 0 \\ 0 \end{bmatrix} \\ &= \boldsymbol{\Gamma}(\mathbf{o}_b)\mathbf{u}_t^b + \boldsymbol{\lambda}(u_x, u_y, u_z, \mathbf{o}_b), \end{aligned} \tag{5}$$

where

$$\boldsymbol{\Gamma}(\mathbf{o}_b) = \text{blockdiag}\left(\text{diag}\left(\left[c_\psi c_\theta, s_\psi s_\theta s_\phi + c_\psi c_\phi, c_\theta c_\phi \right] \right), \mathbf{I}_3 \right),$$

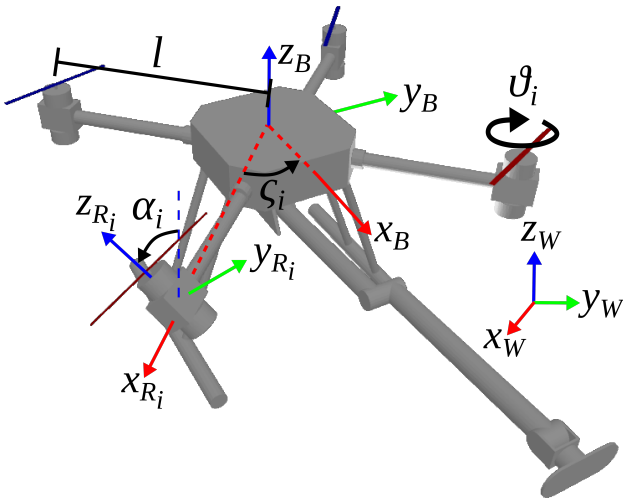

Figure 1: Tilting propellers kinematic parameters.

and $\boldsymbol{\lambda}(u_x, u_y, u_z, \mathbf{o}_b) \in \mathbb{R}^6$ is a disturbance-like vector containing control crossed-terms.

For the quadrotor with tilting propellers considered in this part of the chapter, refer to Figure 1. Each rotor is located at a distance l with an orientation ς with respect to Σ_b. Each propeller can be commanded to tilt of an angle α about its $\mathbf{x}_R$ axis. Hence, the 6-dimensional wrench supplied by the propellers is given by [18]

$$\mathbf{u}_t^b = k_f \begin{bmatrix} \sum_{i=1}^4 \varpi_i^2 \mathbf{R}_{e_3}(\varsigma_i)\mathbf{R}_{e_1}(\alpha_i)\mathbf{e}_3 \\ \sum_{i=1}^4 \left(l\mathbf{R}_{e_3}(\varsigma_i) \times \varpi_i^2 \mathbf{R}_{e_3}(\varsigma_i)\mathbf{R}_{e_1}(\alpha_i)\mathbf{e}_3 - \vartheta_i \sigma \varpi_i^2 \mathbf{R}_{e_3}(\varsigma_i)\mathbf{R}_{e_1}(\alpha_i)\mathbf{e}_3 \right) \end{bmatrix}. \tag{6}$$

The allocation problem is to find the rotors speed ϖ_i and the angles α_i to supply the wrench $\mathbf{u}_t^b$ [19]. The thrust supplied by each rotor is compound by

$$\mathbf{f}_i^b = \begin{bmatrix} f_{i,xy}^b \\ f_{i,z}^b \end{bmatrix} = k_f \varpi_i^2 \begin{bmatrix} \sin(\alpha_i) \\ \cos(\alpha_i) \end{bmatrix} \in \mathbb{R}^2,$$

with $i = 1, \ldots, 4$. Notice that for $\alpha_i = 0,\ i = 1, \ldots, 4$, the thrust is the same as a parallel-axes quadrotor. Furthermore, ς_i determines if the quadrotor is "X" or "+" type. The control thrust of each rotor can be retrieved as follows

$$u_{T,i} = \sqrt{f_{i,xy}^{b2} + f_{i,z}^{b2}} > 0. \tag{7}$$

Consequently, the angular velocity of each rotor is computed through the following expression

$$\varpi_i = \sqrt{\frac{u_{T,i}}{k_f}}, \tag{8}$$

and the tilting angle of each rotor is obtained through

$$\alpha_i = \text{arctan2}\left(f_{i,z}^b, |f_{i,xy}^b|\right), \tag{9}$$

with $i = 1, \ldots, 4$ in all the above expressions.

Then, splitting the static parameters from (6), the following allocation matrix is proposed

$$\mathbf{\Lambda} = \begin{bmatrix} 0 & 0 & 0 & 0 & s_{\varsigma_1} & s_{\varsigma_2} & s_{\varsigma_3} & s_{\varsigma_4} \\ 0 & 0 & 0 & 0 & -c_{\varsigma_1} & -c_{\varsigma_2} & -c_{\varsigma_3} & -c_{\varsigma_4} \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ ls_{\varsigma_1} & ls_{\varsigma_2} & ls_{\varsigma_3} & ls_{\varsigma_4} & s_{\varsigma_1}\vartheta_1\sigma & s_{\varsigma_2}\vartheta_2\sigma & s_{\varsigma_3}\vartheta_3\sigma & s_{\varsigma_4}\vartheta_4\sigma \\ -lc_{\varsigma_1} & -lc_{\varsigma_2} & -lc_{\varsigma_3} & -lc_{\varsigma_4} & -c_{\varsigma_1}\vartheta_1\sigma & -c_{\varsigma_2}\vartheta_2\sigma & -c_{\varsigma_3}\vartheta_3\sigma & -c_{\varsigma_4}\vartheta_4\sigma \\ -\vartheta_1\sigma & -\vartheta_2\sigma & -\vartheta_3\sigma & -\vartheta_4\sigma & -l & -l & -l & -l \end{bmatrix}. \tag{10}$$

Notice that the right pseudo-inverse of such an allocation matrix exists, $\mathbf{\Lambda}^\dagger = \left(\mathbf{\Lambda}^{\mathrm{T}}\mathbf{\Lambda}\right)^{-1}\mathbf{\Lambda}^{\mathrm{T}}$. Therefore, the mapping from the control wrench $\mathbf{u}_t^b$ to the actuators signals is

$$\mathbf{f}^b = \left[\mathbf{f}_1^{b^{\mathrm{T}}}, \mathbf{f}_2^{b^{\mathrm{T}}}, \mathbf{f}_3^{b^{\mathrm{T}}}, \mathbf{f}_4^{b^{\mathrm{T}}}\right]^{\mathrm{T}} = \mathbf{\Lambda}^\dagger \mathbf{u}_t^b \in \mathbb{R}^8. \tag{11}$$

Control design

Consider the omnidirectional quadrotor dynamics (3) with split control crossed-terms as in (5). A control problem for this system is to lead the pose of the UAV to the desired value, despite the disturbances caused by the on-board manipulator motion and by the control crossed-terms. From a mathematically viewpoint, the problem is described as follows

$$\lim_{t\to\infty} \tilde{\boldsymbol{\xi}} \to 0_6,$$
$$\lim_{t\to\infty} \dot{\tilde{\boldsymbol{\xi}}} \to 0_6,$$

where $\tilde{\boldsymbol{\xi}} = \boldsymbol{\xi}^d - \boldsymbol{\xi} \in \mathbb{R}^6$ denotes the pose error, with $\boldsymbol{\xi}^d \in \mathbb{R}^6$ indicating the desired pose that should be at least twice differentiable, and $\dot{\tilde{\boldsymbol{\xi}}} = \dot{\boldsymbol{\xi}}^d - \dot{\boldsymbol{\xi}} \in \mathbb{R}^6$ is the velocity pose error, with $\dot{\boldsymbol{\xi}}^d \in \mathbb{R}^6$ indicating the desired velocity of the pose vector. For the following, $\ddot{\boldsymbol{\xi}}^d \in \mathbb{R}^6$ is the desired acceleration of the UAV's pose.

The following controller is proposed to lead the UAV to the target trajectory

$$\mathbf{u}_t^b = \mathbf{\Gamma}(\mathbf{o}_b)^{-1}\begin{bmatrix} m\mathbf{I}_3 & 0_{3\times3} \\ 0_{3\times3} & \mathbf{M}_b(\mathbf{o}_b) \end{bmatrix}\left[\mathbf{K}_1 \lfloor \mathbf{s} \rceil^{1/2} + \mathbf{K}_2\mathbf{w} + \mathbf{K}_v\dot{\tilde{\boldsymbol{\xi}}} + \ddot{\boldsymbol{\xi}}^d\right] + \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & \mathbf{C}(\mathbf{o}_b, \dot{\mathbf{o}}_b) \end{bmatrix}\dot{\boldsymbol{\xi}}, \tag{12a}$$

$$\dot{\mathbf{w}} = \text{sign}(\mathbf{s}), \tag{12b}$$

where $\mathbf{K}_v,\ \mathbf{K}_1,\ \mathbf{K}_2 \in \mathbb{R}^{6\times6}$ denote constant gains and $\mathbf{s}$ denotes the following sliding surface

$$\mathbf{s} = \mathbf{B}\tilde{\boldsymbol{\xi}} + \dot{\tilde{\boldsymbol{\xi}}}, \tag{13}$$

with $\mathbf{B} \in \mathbb{R}^{6\times6}$ also constant.

In order to deliver the desired force for a long time (*e.g.*, more than 30 seconds), the tracking controller is extended so that the behaviour of the aerial manipulator is as a flying tool able to exert a 6-dimensional wrench [20]. In this case, the experiments are only performed to deliver sustained force along the x_B axis of Σ_b (see Figure 1). Thus, the proposed controller (12) can be extended for force regulation as follows

$$\mathbf{u}_t^b = \boldsymbol{\Gamma}(\mathbf{o}_b)^{-1} \begin{bmatrix} m\mathbf{I}_3 & 0_{3\times3} \\ 0_{3\times3} & \mathbf{M}_b(\mathbf{o}_b) \end{bmatrix} \Big(\mathbf{K}_v\dot{\tilde{\boldsymbol{\xi}}} + \Big(\mathbf{I}_6 - \mathbf{S}_1\Big)\mathbf{K}_1 \lfloor \mathbf{s} \rceil^{1/2} + \Big(\mathbf{I}_6 - \mathbf{S}_1\Big)\mathbf{S}_2\mathbf{K}_2\mathbf{w} \tag{14a}$$

$$+\text{sat}\Big(\mathbf{S}_1\mathbf{K}_f\tilde{\mathbf{f}}\Big)\ \Big) + \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & \mathbf{C}(\mathbf{o}_b, \dot{\mathbf{o}}_b) \end{bmatrix} \dot{\boldsymbol{\xi}}, \tag{14b}$$

$$\dot{\mathbf{w}} = \text{sign}(\mathbf{s}), \tag{14c}$$

where $\tilde{\mathbf{f}} = \mathbf{f}^d - \mathbf{f}^m \in \mathbb{R}^6$ denotes the error between the desired, $\mathbf{f}^d$, and the measured wrench, $\mathbf{f}^m$, at the tip of the end-effector, $\mathbf{K}_f \in \mathbb{R}^{6\times6}$ denotes a constant gain matrix and $\mathbf{S}_1, \mathbf{S}_2 \in \mathbb{R}^{6\times6}$ denote two switching matrices choosing whether or not to use the force regulation and the integral action, respectively.

Validation in the Gazebo simulator

In order to assess the proposed STSMC, the following experiments were performed in Gazebo simulator: (i) trajectory tracking, and (ii) flying to a target pose and applying a sustained force.

The STSMC was programmed in C++ language, and it is executed as a ROS node communicated with Gazebo to obtain the odometry and joint states from the model. Another node commands both the manipulator and the UAV set points. The aerial manipulator used for this work is displayed in Figures 1 and 2. It consists of a quadrotor UAV whose propellers are actuated by a servomotor to rotate about their corresponding x_R axis. The on-board manipulator is also actuated by servomotors and controlled in position. The joint configuration can be appreciated in Figure 2, and the related kinematics is expressed in Σ_b. The weight of the UAV is $9.8kg$, and the arm weights about $1.7kg$.

The first case study to asses the proposed STSMC (see (12)) consists of commanding the

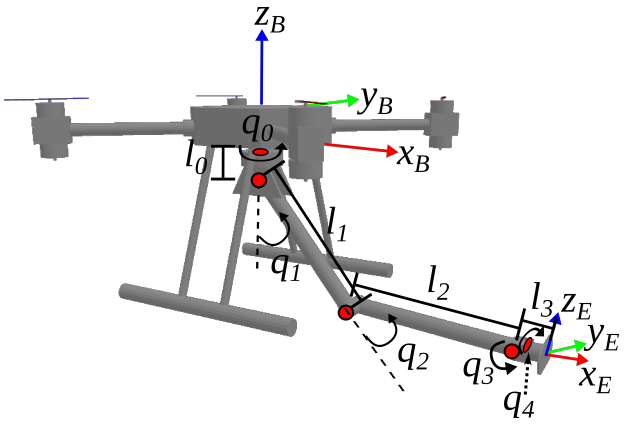

Figure 2: On-board manipulator kinematic parameters.

UAV to track the following trajectory in Cartesian space for 90 seconds

$$\mathbf{p}_b^d = \begin{cases} r\cos(at) \\ r\sin(at) \\ 1 + 0.1t \end{cases}, \quad \dot{\mathbf{p}}_b^d = \begin{cases} -ar\sin(at) \\ ar\cos(at) \\ 0.1 \end{cases}, \quad \ddot{\mathbf{p}}_b^d = \begin{cases} -a^2 r\cos(at) \\ -a^2 r\sin(at) \\ 0 \end{cases}, \tag{15}$$

where $r = 0.4m$ denotes the radius of the circle described mathematically above, and $a = 0.25$ denotes the period of the sinusoidal signals. The desired orientation is $\mathbf{o}_b^d = 0_3$. Figure 3 shows the behaviour of the controlled states. During the first 25 seconds of the experiment, first, the UAV is taken-off. Then, the manipulator is moved to a home setup. Finally, the UAV leads to a target pose close to the beginning of the trajectory.

After this process, the trajectory is commanded for 90 seconds. Figure 3 shows the behaviour of the UAV's position and orientation, where it can be appreciated the satisfactory tracking of the reference trajectory. The average norms of the error vectors are $\|\tilde{\mathbf{p}}_b\| = 0.0086m$ and $\|\tilde{\mathbf{o}}_b\| = 0.1352deg$. Figure 4 shows the motion of the aerial manipulator in the Cartesian space, where it can be observed that the trajectory is correctly tracked. A video of the carried out simulation is available online [1].

In the second case study, the aerial manipulator is commanded to the target Cartesian position $\mathbf{p}_b^d = \begin{bmatrix} 0.7, 0.5, 1.8 \end{bmatrix}^{\mathrm{T}} m$, to apply the desired force of $2.2N$ for at least 60 seconds. The force is measured using a Gazebo contact sensor plugin at the tip of the end-effector of the arm, pointed in Figure 2 as the frame $\Sigma_e : x_E, y_E, z_E$. The experiment also is compound by the stages: (i) take-off, (ii) arm setup (40 seconds), (iii) approach to the point (120 seconds), (iv) contact phase (20 seconds), and (v) force regulation (150 seconds). The last stage involves

[1] https://youtu.be/pJLFsVOek7M

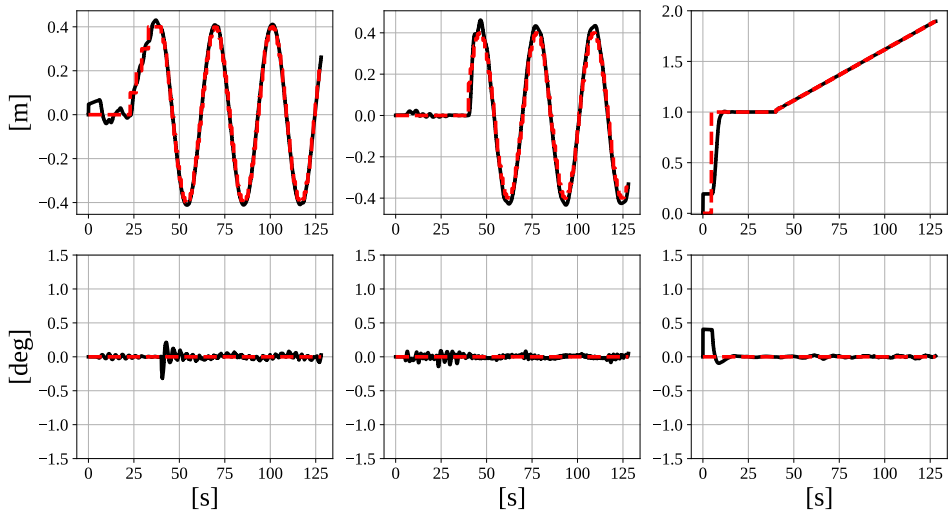

Figure 3: Behaviour of the controlled states during the tracking experiment. Top line: the three components of the UAV's position, $\mathbf{p}_b$. Bottom line: the three components of the UAV's attitude, $\mathbf{o}_b$. - - - Reference. — Measured states.

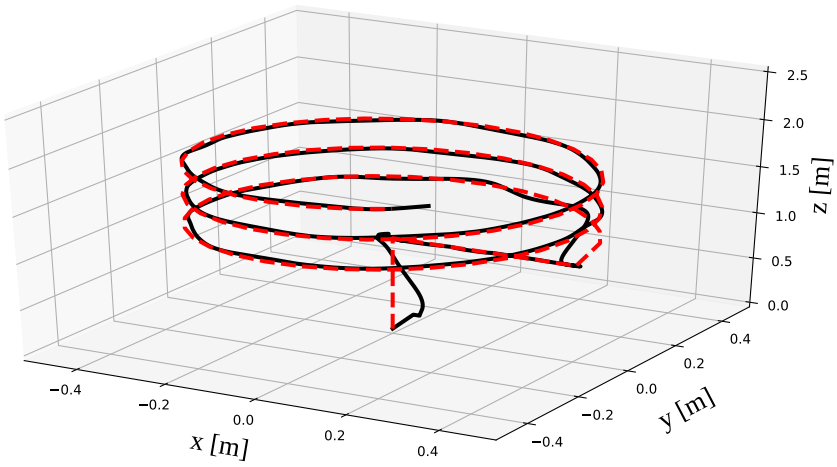

Figure 4: Motion of the aerial manipulator in the Cartesian space. - - - Reference trajectory. — Measured trajectory.

both reaching and then holding the desired force. The sample time for the force regulation is of 1 second. Figure 5 shows the six controlled states of the UAV. The x_B axis is not well regulated; however, a higher reference has been commanded to perform the contact before the force regulation. Nevertheless, the other degrees of freedom are precisely controlled. The results of the force control phase are displayed in Figure 6, where it can be appreciated that the force is successfully regulated to $2.2N$ for more than 60 seconds. The steady-state force error norm is $\|\tilde{F}\| = 0.000012N$. The video of the carried out simulation is available online [2].

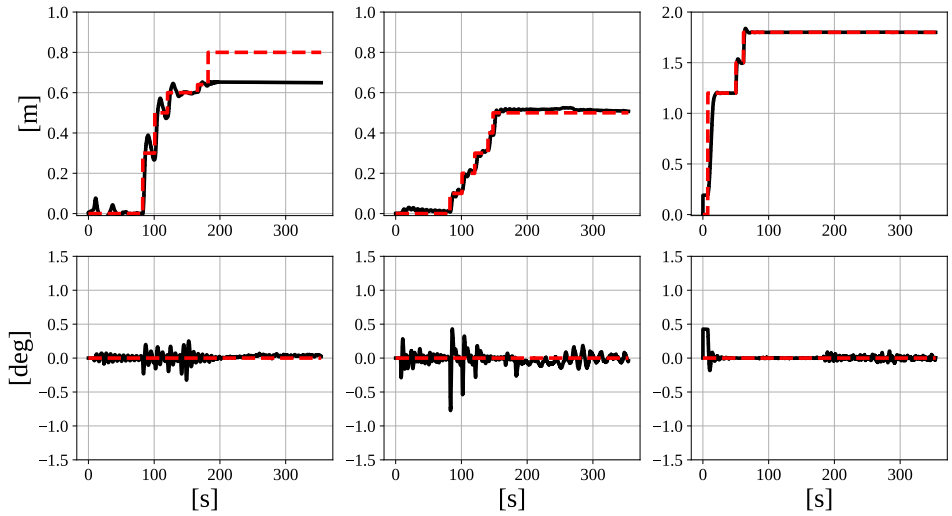

Figure 5: Behaviour of the controlled states during the flight and force regulation experiment. Top line: the three components of the UAV's position, $\mathbf{p}_b$. Bottom line: the three components of the UAV's attitude, $\mathbf{o}_b$. - - - Reference. — Measured states.

Part II - Hardware-in-the-loop simulator for a physical human-aerial manipulator cooperation

The developed hardware-in-the-loop (HIL) simulator architecture consists of three main parts: (i) a model-based simulation of an unmanned aerial manipulator (UAM), composed of parallel-axes quadrotor equipped with a 6-DoF robotic arm; (ii) a hardware interface to enable a force-based interaction with the simulated robotic model while rendering the motion effects of the UAM floating base; and (iii) a bilateral software communication interface connecting the hardware with the simulated model. All these components are integrated through the Robot

[2] https://youtu.be/Ks261f6orMw

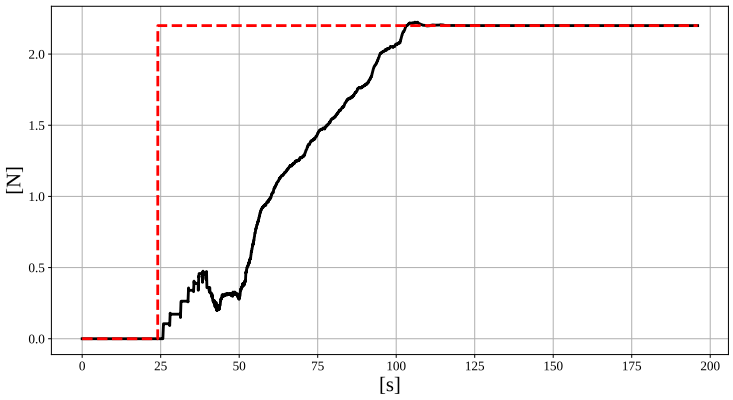

Figure 6: Controlled force along the x_B axis of Σ_b. - - - Reference. — Measured force.

Operating System (ROS) framework, as shown in Figure 7, with specifically designed modules for each of them.

As the hardware interface interacts with the human operator or the environment, all the exchanged forces, measured by a force sensor, are applied to the simulated UAM. The simulation is carried out in Gazebo. The position of the floating base, affected by the interaction forces, is then fed back to the hardware interface, which adjusts its position consequently. This exchange of information between the real and simulated worlds happens simultaneously using standard ROS messages. Although the hardware and simulation controllers have been tailored for the specific experiments, the proposed architecture is general enough to allow the deployment of different control strategies, which the users might develop according to their needs. The code is

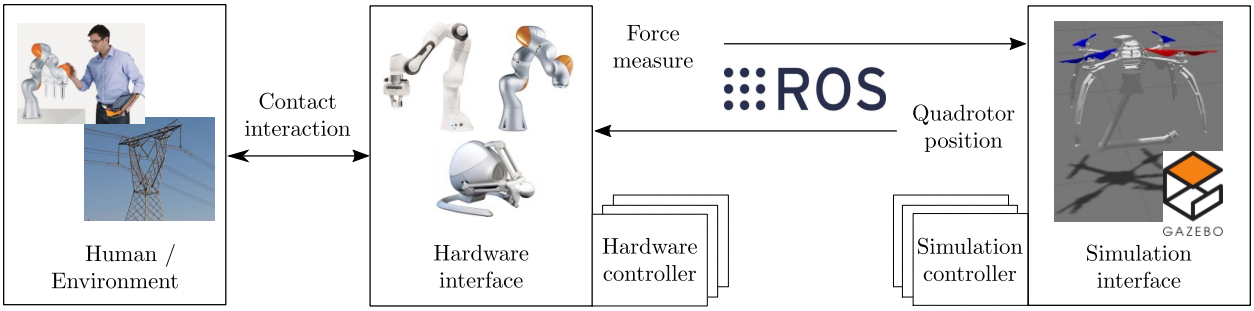

Figure 7: Conceptual scheme of the simulator architecture.

indeed freely released [3].

Related work and contribution

So far, most of the literature regarding human-drone interaction relies on communication through speech [21], gestures [22], brain-computer interfaces [23] and multimodal interaction [24] [25]. An exhaustive overview can be found in [26]. A few works indeed deal with the problem of close physical cooperation between drones and humans [27]. Safe to touch UAVs have been considered in [28] and [29], but the first example of a hardware-in-the-loop simulator for human-UAV interaction was devised in [30]. Here, the human can command a UAV by exchanging forces measured by buttons pushed in a contact point. The interaction forces are then used in an admittance control scheme to modify the vehicle reference trajectory. For similar purposes, an admittance control scheme was also employed in [31]. If in these last works the interaction happened using pushing actions, in [32] a tethered interaction was investigated. Here a UAV pulls the human along the desired path, while the pulling force is used as an indirect communication channel.

This chapter's part extends what has been already presented in [33] by adding the stability proof.

Simulation side

The simulated quadrotor was implemented using the plugins and functionalities offered by the RotorS library [34]. It is modeled as a rigid body in the space actuated by four propellers with parallel axes. The classical parallel-axes quadrotor dynamic model is

$$m\ddot{\mathbf{p}}_b = mg\mathbf{e}_3 - u_T\mathbf{R}_b\mathbf{e}_3 + \mathbf{f}_{ext}, \tag{16a}$$

$$\dot{\mathbf{R}}_b = \mathbf{R}_b\mathbf{S}(\boldsymbol{\omega}_b^b), \tag{16b}$$

$$\mathbf{I}_b\dot{\boldsymbol{\omega}}_b^b = -\mathbf{S}(\boldsymbol{\omega}_b^b)\mathbf{I}_b\boldsymbol{\omega}_b^b + \boldsymbol{\tau}^b + \boldsymbol{\tau}_{ext}^b, \tag{16c}$$

where $\mathbf{I}_b \in \mathbb{R}^{3\times3}$ denotes the constant inertia matrix of the UAV in Σ_b, $\boldsymbol{\omega}_b^b = \mathbf{R}_b^{\mathrm{T}}\omega_b \in \mathbb{R}^3$ denotes the rotation velocity vector of Σ_b with respect to itself, $\mathbf{f}_{ext} \in \mathbb{R}^3$ and $\boldsymbol{\tau}_{ext}^b \in \mathbb{R}^3$ represent the external forces and torques disturbances acting on the quadrotor, respectively. These last take into account unmodeled terms (*e.g.*, aerodynamic disturbances), arm movements, and human-UAM interaction forces.

The allocation problem for the parallel-axes quadrotor can be seen as before with $\alpha_i = 0$, $\forall i = 1, \dots 4$. The total thrust supplied by each rotor is thus only the scalar $f_i, z^b = k_f\varpi_i^2 > 0$.

[3] https://github.com/prisma-lab/HIL_airmanip

The mapping from $\mathbf{u}_f^b$ to $\mathbf{f}_f^b \in \mathbb{R}^4$ is given by $\mathbf{f}_f^b = \mathbf{\Lambda}_f \mathbf{u}_f^b$, with the allocation matrix given by [35]

$$\mathbf{\Lambda}_f = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & l & 0 & -l \\ -l & 0 & l & 0 \\ \sigma & -\sigma & \sigma & -\sigma \end{bmatrix}. \tag{17}$$

The actuated joints of the 6-DoF arm attached to the quadrotor are configured as an anthropomorphic arm with a spherical wrist. Denote with Σ_e the manipulator's end-effector frame while its base frame coincides with Σ_b. The direct kinematics from Σ_b to Σ_e is described by

$$\mathbf{A}_e^b = \begin{bmatrix} \mathbf{R}_e^b(\mathbf{q}) & \mathbf{p}_e^b(\mathbf{q}) \\ 0_3^{\mathrm{T}} & 1 \end{bmatrix}, \tag{18}$$

where $\mathbf{R}_e^b \in SO(3)$ and $\mathbf{p}_e^b \in \mathbb{R}^3$ denote the rotation matrix and the position of Σ_e with respect to Σ_b, respectively.

Hardware

The hardware interface connected to the simulation allows the user to interact with the simulated environment through the measured contact forces. At the same time, the user receives the haptic feedback related to the movements of the simulated UAM, also taking into account the position of the floating base. Notice that it is unnecessary to have a user interacting with the hardware interface: the interaction forces can also come from the environment while testing autonomous control strategies for physical interaction. The architecture is independent of the specific hardware, which can be any device as long as its position in the space can be commanded (*e.g.*, manipulators and/or haptic interfaces). In this work, the hardware interface consists of a 7-DoF KUKA IIWA manipulator equipped with an ATI Mini45 force/torque sensor at the end-effector. Also, since the hardware interface mimics the simulated UAM in operational space coordinates, the two kinematic chains can be structurally different if a suitable inverse kinematics algorithm is adopted.

In particular, to perform a safe human-robot interaction, the dynamics of the hardware interface are given by an admittance controller. Let $\Sigma_{b'}$ denote the base frame of the hardware side. Let $\mathbf{p}_{e'}^{b'} \in \mathbb{R}^3$ denote the position of the hardware end-effector frame, $\Sigma_{e'}$, with respect to $\Sigma_{b'}$, and let $\mathbf{p}_{e',c}^{b'} \in \mathbb{R}^3$ denote the position of the compliant frame, $\Sigma_{e',c}$, with respect to $\Sigma_{b'}$. The hardware-side admittance equations are

$$\mathbf{M}_d \ddot{\tilde{\mathbf{x}}}' + \mathbf{K}_D \dot{\tilde{\mathbf{x}}}' + \mathbf{K}_P \tilde{\mathbf{x}}' = \mathbf{h}_h + \mathbf{h}_q, \tag{19}$$

where $\mathbf{M}_d \in \mathbb{R}^{6\times6}$, $\mathbf{K}_D \in \mathbb{R}^{6\times6}$, and $\mathbf{K}_P \in \mathbb{R}^{6\times6}$ denote the apparent mass, damping, and stiffness matrices, respectively, while $\tilde{\mathbf{x}}' = \left[\tilde{\mathbf{x}}_p'^{\mathrm{T}}, \tilde{\mathbf{x}}_o'^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^6$ is the operational space error between $\Sigma_{e'}$ and $\Sigma_{e',c}$. In particular, $\tilde{\mathbf{x}}_p' = \mathbf{p}_{e'}^{b'} - \mathbf{p}_{e',c}^{b'} \in \mathbb{R}^3$ is the position error, whereas the orientation error $\tilde{\mathbf{x}}_o' \in \mathbb{R}^3$ is given by the vector part of the quaternion representing the rotation between the hardware compliant frame and the hardware end-effector frame, expressed by $\mathbf{R}_{e',c}^{e'} \in SO(3)$. The right-hand side of (19) represents the wrench applied to the system given by: (i) $\mathbf{h}_h \in \mathbb{R}^6$, the interaction wrench exerted by the human operator at the hardware end-effector $\Sigma_{e'}$ expressed in $\Sigma_{b'}$; (ii) $\mathbf{h}_q \in \mathbb{R}^6$, the haptic feedback from the simulated UAM detailed in the next section.

Communication interface

The wrench, $\mathbf{h}_e \in \mathbb{R}^6$, applied in simulation at the UAM's end-effector can be then computed as $\mathbf{h}_e = \mathbf{Ad}_{T_{ew}}\,\mathbf{Ad}_{T_{wb'}}\,\mathbf{Ad}_{T_{b'e'}}\mathbf{h}_h$,where $\mathbf{Ad}_{T_{12}} \in \mathbb{R}^{6\times6}$ denotes the adjoint transformation matrix between two generic frames, Σ_1 and Σ_2 [36]. Notice that the transformation matrix $\boldsymbol{T}_{b'}^{w}$ should be chosen by the user to connect the real hardware base frame to the simulated world.

Regarding the simulation feedback to the hardware interface, the displacement $\mathbf{e}_p \in \mathbb{R}^3$ between the commanded and the actual position of the quadrotor's CoM can be fed back onto the hardware interface to emulate the effects of the floating base displacements. In particular, this contribution can be seen as an additional wrench, playing the role of haptic feedback for the human operator, given by

$$\mathbf{M}_d\ddot{\mathbf{e}}_p' + \mathbf{K}_D\dot{\mathbf{e}}_p' + \mathbf{K}_P\mathbf{e}_p' = \mathbf{h}_q\,, \tag{20}$$

where $\mathbf{e}_p' = \mathbf{Ad}_{T_{b'w}}\mathbf{e}_p$ denotes the quadrotor's displacement transformed from Σ_w into $\Sigma_{b'}$.

System controllers

Here we present the hardware and the simulation controllers later used in the two case studies. Although the simulator itself is not strictly related to the type of controller, in our case, a decentralized controller [2] has been implemented on the simulated UAM, employing a geometric position tracking controller for the quadrotor and a Cartesian variable admittance controller for the arm. Also, the quadrotor's position tracking is enhanced with a momentum-based external wrench estimator [17] to compensate for the arm dynamics, the interaction forces, and other unmodeled disturbances.

Because of the under-actuation of the system, a hierarchical approach is followed to control both the position, $\mathbf{p}_b$, and the attitude, $\mathbf{R}_b$, of the quadrotor. From this point of view, the geometric tracking controller in $SE(3)$ [37] is implemented on the quadrotor.

The outer position loop tracking errors are $\mathbf{e}_p = \mathbf{p}_b - \mathbf{p}_b^d$ and $\mathbf{e}_v = \dot{\mathbf{p}}_b - \dot{\mathbf{p}}_b^d$, where $\mathbf{p}_b^d \in \mathbb{R}^3$ denotes the desired position of the Σ_b's origin in Σ_w, obtained from an external trajectory planner. Let $\mathbf{R}_{b,d} = [\mathbf{x}_{b,d}, \mathbf{y}_{b,d}, \mathbf{z}_{b,d}] \in SO(3)$ denote the desired rotation matrix, where $x_{b,d} \in \mathbb{R}^3$ is given from the trajectory planner. The tracking errors of the inner attitude loop are given by $\mathbf{e}_R = 0.5\,(\mathbf{R}_{b,d}^{\mathrm{T}}\,\mathbf{R}_b - \mathbf{R}_b^{\mathrm{T}}\,\mathbf{R}_{b,d})^{\vee}$ and $\mathbf{e}_\omega = \boldsymbol{\omega}_b^b - \mathbf{R}_b^{\mathrm{T}}\,\mathbf{R}_{b,d}\,\boldsymbol{\omega}_{b,d}^{b,d}$, where $\boldsymbol{\omega}_{b,d}^{b,d} \in \mathbb{R}^3$ the desired body rotation velocity in Σ_b, and $^{\vee} : \mathbb{R}^{3\times3} \to \mathbb{R}^3$ is a map performing the inverse of the skew-symmetric operator.

Theorem .1. *In the absence of external disturbances, or in the case they are negligible, the necessary thrust, u_T, the desired body axis, $\mathbf{z}_{b,d} \in R^3$, and the attitude control law, $\boldsymbol{\tau}^b$, that asymptotically bring to zero the outer position loop and the inner attitude loop tracking errors be computed as*

$$u_T = (\mathbf{K}_p\,\mathbf{e}_p + \mathbf{K}_v\mathbf{e}_v + mg\mathbf{e}_3 - m\ddot{\mathbf{p}}_{b,d})^{\mathrm{T}}\mathbf{R}_b\mathbf{e}_3, \tag{21a}$$

$$\mathbf{z}_{b,d} = -\frac{-\mathbf{K}_p\mathbf{e}_p - \mathbf{K}_v\mathbf{e}_v - mg\mathbf{e}_3 + m\ddot{\mathbf{p}}_b^d}{\| -\mathbf{K}_p\mathbf{e}_p - \mathbf{K}_v\mathbf{e}_v - mg\mathbf{e}_3 + m\ddot{\mathbf{p}}_{b,d}\|}, \tag{21b}$$

where $\mathbf{K}_p \in \mathbb{R}^{3\times3}$ and $\mathbf{K}_v \in \mathbb{R}^{3\times3}$ denote symmetric positive definite gain matrices, and

$$\boldsymbol{\tau}^b = -\mathbf{K}_R\,\mathbf{e}_R - \mathbf{K}_\omega\,\mathbf{e}_\omega + S(\boldsymbol{\omega}_b^b)\,\mathbf{I}_b\,\boldsymbol{\omega}_b^b - \mathbf{I}_b\,(\mathbf{S}(\boldsymbol{\omega}_b^b)\,\mathbf{R}_b^{\mathrm{T}}\,\mathbf{R}_{b,d}\,\boldsymbol{\omega}_{b,d}^{b,d} - \mathbf{R}_b^{\mathrm{T}}\,\mathbf{R}_{b,d}\,\dot{\boldsymbol{\omega}}_{b,d}^{b,d}), \tag{22}$$

with $\mathbf{K}_R \in \mathbb{R}^{3\times3}$ and $\mathbf{K}_\omega \in \mathbb{R}^{3\times3}$ symmetric positive definite gain matrices.

Proof. See [37]. □

Because there might be significant external disturbances in our case, this control scheme needs an external wrench estimator to keep good performance in trajectory tracking. In this work, the estimator presented in [17] has been used.

Due to the limited payload capabilities of aerial platforms, the arm of a UAM is typically actuated by position- or velocity-controlled joints (*e.g.*, servo motors) [2]. In this case, admittance control can guarantee compliance and safety during the interaction with both environment and humans. However, as human and environment interaction tasks require different compliance choices, a variable-gain admittance control is used. In the operational space, the admittance controlled manipulator dynamics is given by

$$\mathbf{M}_d\ddot{\tilde{\mathbf{x}}} + \mathbf{K}_D\dot{\tilde{\mathbf{x}}} + \mathbf{K}_P\tilde{\mathbf{x}} = \mathbf{h}_e^b, \tag{23}$$

where $\tilde{\mathbf{x}} = \left[\tilde{\mathbf{x}}_p^{\mathrm{T}}, \tilde{\mathbf{x}}_o^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^6$ denotes the operational space error between the desired end-effector frame and the compliant frame, in the simulation side, while $\mathbf{h}_e^b \in \mathbb{R}^6$ is the wrench measured at the simulated end-effector, expressed in Σ_b. In particular, $\tilde{\mathbf{x}}_p = \mathbf{p}_e^b - \mathbf{p}_{e,c}^b \in \mathbb{R}^3$ is the position

error, whereas the orientation error $\tilde{\mathbf{x}}_o \in \mathbb{R}^3$ is given by the vector part of the quaternion representing the rotation between the simulated compliant frame and the simulated end-effector frame, expressed by $\mathbf{R}^e_{e,c} \in SO(3)$.

The dynamics in (23) are passive[4] with respect to the power port $(\mathbf{h}^b_e, \dot{\tilde{\mathbf{x}}})$. Indeed, by choosing as a storage function

$$V(\tilde{\mathbf{x}}, \dot{\tilde{\mathbf{x}}}) = \frac{1}{2}\dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{M}_d\dot{\tilde{\mathbf{x}}} + \frac{1}{2}\tilde{\mathbf{x}}^{\mathrm{T}}\mathbf{K}_P\tilde{\mathbf{x}} \,, \tag{24}$$

its time derivative is

$$\dot{V} = \dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{M}_d\ddot{\tilde{\mathbf{x}}} + \dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{K}_P\tilde{\mathbf{x}} = \dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{h}^b_e - \dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{K}_D\dot{\tilde{\mathbf{x}}} \leq \dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{h}^b_e. \tag{25}$$

The same passivity argument can be extended to the hardware side (see (19)) with respect to the interaction force $\mathbf{h}_h$. By defining $\bar{\mathbf{x}} = \tilde{\mathbf{x}} - \boldsymbol{e}'_p$, equation (19) can be rewritten as $\mathbf{M}_d\ddot{\bar{\mathbf{x}}} + \mathbf{K}_D\dot{\bar{\mathbf{x}}} + \mathbf{K}_P\bar{\mathbf{x}} = \mathbf{h}_h$, which is passive with respect to the power port $(\boldsymbol{h}_h, \dot{\bar{\mathbf{x}}})$ with storage function

$$V(\bar{\mathbf{x}}, \dot{\bar{\mathbf{x}}}) = \frac{1}{2}\dot{\bar{\mathbf{x}}}^{\mathrm{T}}\mathbf{M}_d\dot{\bar{\mathbf{x}}} + \frac{1}{2}\bar{\mathbf{x}}^{\mathrm{T}}\mathbf{K}_P\bar{\mathbf{x}} \,. \tag{26}$$

However, if the admittance gains are time-variant, functions in (24) and (26) are no longer valid. Indeed, equation (25) becomes

$$\begin{aligned} \dot{V} &= \dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{M}_d\ddot{\tilde{\mathbf{x}}} + \dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{K}_P\tilde{\mathbf{x}} + \frac{1}{2}\left[\dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\dot{\mathbf{M}}_d\dot{\tilde{\mathbf{x}}} + \tilde{\mathbf{x}}^{\mathrm{T}}\dot{\mathbf{K}}_P\tilde{\mathbf{x}}\right] \\ &= \dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{h}_e - \dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{K}_D\dot{\tilde{\mathbf{x}}} + \frac{1}{2}\left[\dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\dot{\mathbf{M}}_d\dot{\tilde{\mathbf{x}}} + \tilde{\mathbf{x}}^{\mathrm{T}}\dot{\mathbf{K}}_P\tilde{\mathbf{x}}\right] \,, \end{aligned} \tag{27}$$

and passivity is guaranteed only if the following holds

$$\dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{K}_D\dot{\tilde{\mathbf{x}}} \;\geq\; \frac{1}{2}\left[\dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\dot{\mathbf{M}}_d\dot{\tilde{\mathbf{x}}} + \tilde{\mathbf{x}}^{\mathrm{T}}\dot{\mathbf{K}}_P\tilde{\mathbf{x}}\right] \,. \tag{28}$$

Notice that the same can be applied to (26). To guarantee passivity despite (28), an *energy-tank* can be employed [40], [41]. The tank dynamics is $\dot{z} = \frac{\varphi}{z}P_d - \frac{1}{z}\gamma w$, where $\varphi \in \mathbb{R}$ and $\gamma \in R$ denote two parameters, $P_d = \dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\mathbf{K}_D\dot{\tilde{\mathbf{x}}} \geq 0$ denotes the power dissipated by the admittance system (23), $w = \frac{1}{2}\dot{\tilde{\mathbf{x}}}^{\mathrm{T}}\dot{\mathbf{M}}_d\dot{\tilde{\mathbf{x}}} + \tilde{\mathbf{x}}^{\mathrm{T}}\dot{\mathbf{K}}_P\tilde{\mathbf{x}}$ denotes the tank input. Let define the tank's storage

[4]The passivity concept can be revised in [38], [39].

function as $\mathcal{T}(z) = 0.5z^2$. To guarantee the passivity, γ and φ are chosen as

$$\gamma = \begin{cases} 0, & \text{if } \mathcal{T} \geq \overline{\mathcal{T}} \;\&\; w \leq 0 \\ \iota, & \text{otherwise} \end{cases} \tag{29a}$$

$$\iota = \begin{cases} 1, & \text{if } w > 0 \\ \dfrac{1}{2}\left(1 - \cos\left(\pi \dfrac{\mathcal{T} - \underline{\mathcal{T}}}{\overline{\mathcal{T}} - \underline{\mathcal{T}}}\right)\right), & \text{otherwise} \end{cases} \tag{29b}$$

$$\varphi = \begin{cases} \varphi_d, & \text{if } \mathcal{T} < \overline{\mathcal{T}} \\ 0, & \text{otherwise} \end{cases}, \tag{29c}$$

with $\varphi_d \leq 1$ represents the amount of the dissipated energy redirected to the tank, while $\overline{\mathcal{T}} \in \mathbb{R}$ and $\underline{\mathcal{T}} \in \mathbb{R}$ are the upper and lower energy limits of the tank, respectively.

If the desired values of $\dot{\mathbf{M}}_d$ and $\dot{\mathbf{K}}_P$ are substituted by $\iota\dot{\mathbf{M}}_d$ and $\iota\dot{\mathbf{K}}_P$, respectively, the overall storage function is given by

$$\begin{aligned} \dot{V} + \dot{\mathcal{T}} &= \dot{\tilde{\mathbf{x}}}^{\mathrm{T}} \mathbf{M}_d \ddot{\tilde{\mathbf{x}}} + \dot{\tilde{\mathbf{x}}}^{\mathrm{T}} \mathbf{K}_P \tilde{\mathbf{x}} + \iota w + \varphi P_d - \gamma w \\ &= \dot{\tilde{\mathbf{x}}}^{\mathrm{T}} \mathbf{h}_e - \dot{\tilde{\mathbf{x}}}^{\mathrm{T}} \mathbf{K}_D \dot{\tilde{\mathbf{x}}} + \iota w + \varphi P_d - \gamma w \\ &= \dot{\tilde{\mathbf{x}}}^{\mathrm{T}} \mathbf{h}_e - (1 - \varphi) P_d + (\iota - \gamma) w \leq \dot{\tilde{\mathbf{x}}}^{\mathrm{T}} \mathbf{h}_e \,, \end{aligned} \tag{30}$$

where the passivity of arm plus tank system is always verified despite the sign of w.

The energy tank partially stores the energy dissipated by the admittance dynamics, releasing it later if necessary. By injecting controlled amounts of energy in the system, the tank allows performing actions that would typically violate the system passivity, like changing the admittance virtual stiffness or mass.

Since the arm and the quadrotor are controlled separately, using an external wrench estimator to compensate for the arm movements, among the various disturbances, can enhance the stability properties of the quadrotor controller. A momentum-based estimator is used [42], where the external wrench estimation is denoted by $\hat{\boldsymbol{\eta}}_{ext} = \left[\hat{\mathbf{f}}_{ext}^{\mathrm{T}}, \hat{\boldsymbol{\tau}}_{ext}^{b^{\mathrm{T}}}\right]^{\mathrm{T}} \in \mathbb{R}^6$.

Lemma .2. *To effectively perform the compensation in the quadrotor control law, the force estimation, $\hat{\mathbf{f}}_{ext}$, is added to the term $\mathbf{K}_p\,\mathbf{e}_p + \mathbf{K}_v\mathbf{e}_v + mg\mathbf{e}_3 - m\ddot{\mathbf{p}}_{b,d}$ within* (21)*, while the torque estimation, $\hat{\boldsymbol{\tau}}_{ext}^{b}$, is added to* (22)*. The estimation of the external wrench is carried out through*

the following estimator

$$\begin{aligned}\hat{\boldsymbol{\eta}}_{ext}(t) =& \boldsymbol{K}_1 \left(\int_0^t -\hat{\boldsymbol{\eta}}_{ext}(\sigma) + \boldsymbol{K}_2 \Bigg(\boldsymbol{\kappa}(\sigma) \right. \\ & \left. - \int_0^t \left(\begin{bmatrix} u_T \boldsymbol{R}_b \boldsymbol{e}_3 - mg\boldsymbol{e}_3 \\ \boldsymbol{\tau}^b - \boldsymbol{S}(\boldsymbol{\omega}_b^b)\boldsymbol{I}_b\boldsymbol{\omega}_b^b \end{bmatrix} + \hat{\boldsymbol{\eta}}_{ext}(\sigma) \right) d\sigma \Bigg) d\sigma \right), \end{aligned} \tag{31}$$

where $\boldsymbol{\kappa} \in \mathbb{R}^6$ *is the generalised momentum of the system* (16) *defined as* $\boldsymbol{\kappa} = \begin{bmatrix} m\mathbf{I}_3 & 0_{3\times3} \\ 0_{3\times3} & \mathbf{I}_b \end{bmatrix} \begin{bmatrix} \dot{\mathbf{p}}_b \\ \boldsymbol{\omega}_b^b \end{bmatrix}$, *and the matrices* $\mathbf{K}_1 \in \mathbb{R}^{6\times6}$ *and* $\mathbf{K}_2 \in \mathbb{R}^{6\times6}$ *are symmetric positive-definite gains.*

Proof. The geometric flight controller for the UAV and the admittance-controlled arm have been proved to possess stability properties in many past works. However, their combination, especially with a momentum-based external wrench estimator, is not guaranteed to maintain the same properties.

This subsection aims to demonstrate that the dynamics of the UAM are marginally stable and bounded in the presence of unknown and non-vanishing (but still bounded) external force disturbances. In the following, only the boundedness of the linear dynamics will be proven for the sake of brevity, but notice that the effects of the angular dynamics are still taken into account in the UAV dynamics. In this regard, consider both the manipulator and the UAV commanded to be still in the air at the desired position. The admittance-controlled dynamics of the arm with respect to the UAV base frame and the dynamics of the UAV base frame with respect to the world frame can be written, respectively, as

$$\mathbf{M}_e \ddot{\mathbf{p}}_e^b + \mathbf{D}_e \dot{\mathbf{p}}_e^b + \mathbf{K}_e \tilde{\mathbf{p}}_e^b = \mathbf{f}_h, \tag{32a}$$

$$\mathbf{M}_b \ddot{\mathbf{p}}_b + \mathbf{D}_b \dot{\mathbf{p}}_b + \mathbf{K}_b \tilde{\mathbf{p}}_b = \boldsymbol{\delta} + \mathbf{e}_F, \tag{32b}$$

where $\boldsymbol{\delta} \in \mathbb{R}^3$ are the nonlinear coupling dynamics given by the underactuation of the UAV, $\mathbf{f}_h \in R^3$ are the human interaction forces and $\mathbf{e}_F \in R^3$ is the residual of external forces acting on the UAV not instantaneously compensated by the external wrench estimator in (31). Fusing together the two dynamics in (32) yields the arm dynamics into the world frame, Σ_w, whose expression is

$$\begin{aligned}\mathbf{M}_e \ddot{\mathbf{p}}_e + \mathbf{D}_e \dot{\mathbf{p}}_e + \mathbf{K}_e \tilde{\mathbf{p}}_e &= \mathbf{f}_h + \mathbf{M}_e \mathbf{M}_b^{-1}(\boldsymbol{\delta} + \mathbf{e}_F) + (\mathbf{D}_e - \mathbf{M}_e \mathbf{M}_b^{-1} \mathbf{D}_b)\dot{\mathbf{p}}_b + (\mathbf{K}_e - \mathbf{M}_e \mathbf{M}_b^{-1} \mathbf{K}_b)\tilde{\mathbf{p}}_b \\ &= \mathbf{f}_h + \mathbf{M}(\boldsymbol{\delta} + \mathbf{e}_F) + \mathbf{D}\dot{\mathbf{p}}_b + \mathbf{K}\tilde{\mathbf{p}}_b. \end{aligned} \tag{33}$$

Starting from (33), the stability proof is split into two parts: first, the UAV dynamics will be proven to be bounded; then, the manipulator's dynamic into the world frame will be established to be bounded by exploiting the first result.

Consider the non-perturbed UAV dynamics

$$\boldsymbol{\alpha}_b(\bar{\mathbf{x}}_b) = \mathbf{M}_b\ddot{\mathbf{p}}_b + \mathbf{D}_b\dot{\mathbf{p}}_b + \mathbf{K}_b\tilde{\mathbf{p}}_b = 0, \tag{34}$$

where $\dot{\bar{\mathbf{x}}}_b = \left[\dot{\mathbf{p}}_b^{\mathrm{T}}, \tilde{\mathbf{p}}_b^{\mathrm{T}}\right]^{\mathrm{T}}$. Since $\mathbf{M}_b$, $\mathbf{D}_b$, and $\mathbf{K}_b$ are symmetric positive definite matrices, the closed-loop dynamics is asymptotically stable and it can be rewritten in the form $\dot{\bar{\mathbf{x}}}_b = \mathbf{A}_b\bar{\mathbf{x}}_b$, where $\mathbf{A}_b \in \mathbb{R}^{6\times 6}$ can be easily extracted from (34).

From the proven stability of the previous closed-loop system, it follows that the Lyapunov function $V_b(t, \bar{\mathbf{x}}_b) = \frac{1}{2}\bar{\mathbf{x}}_b^{\mathrm{T}}\mathbf{P}_b\bar{\mathbf{x}}_b$ exists and the following Lyapunov equation

$$\mathbf{P}_b\mathbf{A}_b + \mathbf{A}_b^{\mathrm{T}}\mathbf{P}_b = -\mathbf{Q}_b, \tag{35}$$

is satisfied with a unique symmetric and positive definite matrix $\mathbf{P}_b \in \mathbb{R}^{6\times 6}$ given any symmetric and positive definite $\mathbf{Q}_b \in \mathbb{R}^{6\times 6}$. Then, it is easy to show that there exist four scalars $\gamma_1 = \underline{\lambda}_{\mathbf{P}_b}$, $\gamma_2 = \bar{\lambda}_{\mathbf{P}_b}$, $\gamma_3 = \underline{\lambda}_{\mathbf{Q}_b}$ and $\gamma_4 = 2\bar{\lambda}_{\mathbf{P}_b}$ such to satisfy the inequalities

$$\gamma_1\|\bar{\mathbf{x}}_b\|^2 \leq V_b(\bar{\mathbf{x}}_b) \leq \gamma_2\|\bar{\mathbf{x}}_b\|^2, \tag{36a}$$

$$\frac{\partial V_b}{\partial t} + \frac{\partial V_b}{\partial \bar{\mathbf{x}}_b}\dot{\bar{\mathbf{x}}}_b \leq -\gamma_3\|\bar{\mathbf{x}}_b\|^2, \tag{36b}$$

$$\left\|\frac{\partial V_b}{\partial \bar{\mathbf{x}}_b}\right\| \leq \gamma_4\|\bar{\mathbf{x}}_b\|^2, \tag{36c}$$

where $\underline{\lambda}_{\mathbf{X}}$ and $\bar{\lambda}_{\mathbf{X}}$ denote the minimum and maximum eigenvalues of a generic matrix $\mathbf{X}$, respectively.

So far, we have shown a few properties of the non-perturbed UAV dynamics in (34), but these actually have a possibly non-vanishing perturbation on the right side given by $\mathbf{g}_b(\bar{\mathbf{x}}_b, t) = \boldsymbol{\delta} + \mathbf{e}_F$. By recalling the properties and the bounds of the nonlinear coupling term $\boldsymbol{\delta}$ (see [37] for the details), it can be shown that the perturbation term is bounded by

$$\begin{aligned}\|\mathbf{g}_b(\bar{\mathbf{x}}_b, t)\| \leq \|\boldsymbol{\delta}\| + \|\mathbf{e}_F\| &\leq \alpha(\sqrt{2}max\{\bar{\lambda}_{\mathbf{K}_b}, \bar{\lambda}_{\mathbf{D}_b}\}\|\bar{\mathbf{x}}_b\| + B) + B_1 \\ &= \Gamma_1\|\bar{\mathbf{x}}_b\| + \Gamma_2,\end{aligned} \tag{37}$$

with $\alpha < 1$, $\|-mg\mathbf{e}_3 + m\ddot{\mathbf{p}}_{b,d}\| < B$ and a scalar $B > 0$.

By virtue of Lemma 5.2 in [43], if the inequality

$$\Gamma_1 < \frac{\underline{\lambda}_{\mathbf{P}_b}\underline{\lambda}_{\mathbf{Q}_b}}{2\bar{\lambda}_{\mathbf{P}_b}^2} \tag{38}$$

holds, then the UAV dynamics $\bar{\mathbf{x}}_b(t)$ is bounded by

$$\|\bar{\mathbf{x}}_b(t)\| \leq B_2, \quad \forall t \geq t_0, \tag{39}$$

with $B_2 = max\left\{\dfrac{\bar{\lambda}_{\mathbf{P}_b}}{\underline{\lambda}_{\mathbf{P}_b}}\|\bar{\mathbf{x}}_b\|\,,\ \dfrac{\Gamma_2}{\rho}\right\}$ and $\rho = \dfrac{1}{2}\left(\dfrac{\underline{\lambda}_{\mathbf{Q}_b}}{\bar{\lambda}_{\mathbf{P}_b}} - \Gamma_1\dfrac{2\bar{\lambda}_{\mathbf{P}_b}}{\underline{\lambda}_{\mathbf{P}_b}}\right)$. The fulfilment of inequality (38) will be discussed later.

As already done for the UAV dynamics, consider the non-perturbed manipulator dynamics (only for the position part) in the world frame, Σ_w, given by

$$\boldsymbol{\alpha}_e(\bar{\mathbf{x}}_e) = \mathbf{M}_e\ddot{\mathbf{p}}_e + \mathbf{D}_e\dot{\mathbf{p}}_e + \mathbf{K}_e\tilde{\mathbf{p}}_e, \tag{40}$$

where $\dot{\bar{\mathbf{x}}}_e = \left[\dot{\mathbf{p}}_e^{\mathrm{T}}, \tilde{\mathbf{p}}_e^{\mathrm{T}}\right]^{\mathrm{T}}$. As in the previous case, the dynamics is asymptotically stable and it can be written in the form $\dot{\bar{\mathbf{x}}}_e = \mathbf{A}_e\bar{\mathbf{x}}_e$, with $\mathbf{A}_e \in \mathbb{R}^{6\times6}$. Again, the Lyapunov function $V_e(t,\bar{\mathbf{x}}_e) = \dfrac{1}{2}\bar{\mathbf{x}}_e^{\mathrm{T}}\mathbf{P}_e\bar{\mathbf{x}}_e$ exists such as to satisfy (35) with the matrices $\mathbf{A}_e$, $\mathbf{P}_e \in \mathbb{R}^{6\times6}$, and $\mathbf{Q}_e \in \mathbb{R}^{6\times6}$. Accordingly, four new scalars $\gamma_1 = \underline{\lambda}_{\mathbf{P}_e}$, $\gamma_2 = \bar{\lambda}_{\mathbf{P}_e}$, $\gamma_3 = \underline{\lambda}_{\mathbf{Q}_e}$ and $\gamma_4 = 2\bar{\lambda}_{\mathbf{P}_e}$ can be found to satisfy inequalities as in (36) similarly.

Now, consider the perturbation term $\mathbf{g}_e(\bar{\mathbf{x}}_b, t) = \mathbf{f}_h + \mathbf{M}(\boldsymbol{\delta} + \mathbf{e}_F) + \mathbf{D}\dot{\mathbf{p}}_b + \mathbf{K}\tilde{\mathbf{p}}_b$ to the previous nominal system. From (37) and (39), it comes out the boundedness of the previous non-vanishing perturbation

$$\|\mathbf{g}_e(\bar{\mathbf{x}}_b, t)\| \leq B_4 + \bar{\lambda}_{\mathbf{M}}(B_3 + B_1) + \sqrt{2}max\{\bar{\lambda}_{\mathbf{K}}, \bar{\lambda}_{\mathbf{D}}\}B_2 = \Delta_e. \tag{41}$$

According to Lemma 5.2 in [43], since the perturbation term is bounded, for all $\|\bar{\mathbf{x}}_e(t_e)\| < \infty$, there exists a time instant $t_e > 0$ such that

$$\|\bar{\mathbf{x}}_e(t)\| \leq \xi_e e^{-\rho_e(t-t_0)}\|\bar{\mathbf{x}}_e(t_0)\|, \quad \forall t_0 \leq t \leq t_e, \tag{42a}$$

$$\|\bar{\mathbf{x}}_e(t)\| \leq B_5, \quad \forall t \geq t_e, \tag{42b}$$

with

$$\xi_e = \sqrt{\frac{\bar{\lambda}_{\mathbf{P}_e}}{\underline{\lambda}_{\mathbf{P}_e}}}\ ,\ \rho_e = \frac{(1-\epsilon)\underline{\lambda}_{Q_e}}{2\bar{\lambda}_{\mathbf{P}_e}}\ ,\ B_5 = \frac{2\Delta_e\bar{\lambda}_{\mathbf{P}_e}}{\epsilon\,\underline{\lambda}_{\mathbf{Q}_e}}\xi_e\ ,\ \epsilon < 1. \tag{43}$$

From (45), the boundedness of the manipulator dynamics yields

$$\|\bar{\mathbf{x}}_e(t)\| \leq B_6 = max\{\xi_e\|\bar{\mathbf{x}}_e(t_0)\|\,,\ B_5\}, \quad \forall t \geq t_e. \tag{44}$$

In the previous subsection, we have demonstrated that the UAV dynamics and, consequently, the whole UAM dynamics are bound to external non-vanishing forces. However, the entire demonstration is based on the validity of the inequality (38). Since it is not straightforward to demonstrate it theoretically, a numerical way is presented here. In particular, it can be shown from (34) and (35) that the fulfilment of (38) reduces to solving the system

$$\mathbf{A}_b = \begin{bmatrix} -\mathbf{M}_b^{-1}\mathbf{D}_b & -\mathbf{M}_b^{-1}\mathbf{K}_b \\ \\ \mathbf{I}_{3\times3} & 0_{3\times3} \end{bmatrix}, \tag{45a}$$

$$\mathbf{P}_b\mathbf{A}_b + \mathbf{A}_b^{\mathrm{T}}\mathbf{P}_b = -\mathbf{Q}_b, \tag{45b}$$

$$\alpha\sqrt{2}\max\left\{\bar{\lambda}_{\mathbf{K}_b}, \bar{\lambda}_{\mathbf{D}_b}\right\} < \frac{\underline{\lambda}_{\mathbf{Q}_b}\underline{\lambda}_{\mathbf{P}_b}}{2\bar{\lambda}^2_{\mathbf{P}_b}}, \tag{45c}$$

as function of the controller parameters, the system mass, and the initial attitude error α of the geometric controller defined in [37].

The symbolic computations to verify the existence of bounds have been performed in Mathematica [44]. To provide an answer in a reasonable time, the search of solutions has been restricted to the subset in which $\mathbf{Q}_b = \boldsymbol{I}_3$. Notice that this is not a restrictive hypothesis: indeed, the existence of a solution in this subset implies the presence of a solution in the larger subset given by all the symmetric and positive definite matrices $\mathbf{Q}_b$. The result of the computations is visible in Figure 8. The plots show the maximum initial attitude error tracking value for the system's state to be bounded as a function of the UAM mass and its controller parameters. It is interesting to notice that, regardless of the value of the external interaction forces that will be applied to the system, the response of the UAM will always be bounded as long as the force is bounded and our system parameters respect the limits in the graph. Also, by employing a trajectory planner for the UAM, the initial attitude error will usually be very low, ideally zero, making it easy to fall in the cases depicted in the graphs. Moreover, as stated earlier, this is a result obtained in the particular case of $\mathbf{Q}_b = \boldsymbol{I}_3$. It is likely that by considering other values for $\mathbf{Q}_b$, the parameter set that allows the boundedness will enlarge even more.

□

Case study

The effectiveness of the proposed system is evaluated along with two case studies: (i) a collaborative experiment where the human operator attaches a tool to the robot end-effector; and (ii) an autonomous bird diverter installation task. The simulations have been performed on a standard Ubuntu 18.04 distribution with ROS Melodic, running at 200 Hz.

The first case study considers the collaboration task between a human operator and the UAM through the hardware interface. In this context, the aerial platform awaits in mid-air until the operator approaches. The interaction phase starts with the admittance-controlled hardware interface to mount a tool on the manipulator gripper. While in use, the hardware interface provides a compliant behaviour and transfers all the measured forces to the simulated UAM.

When the aerial platform recognises that both the tool is mounted on the hardware interface and the operator has finished the interaction with it, the experiment ends.

The interaction between the human operator and the simulation framework is demonstrated in Figure 9 by the motion of the aerial manipulator $\boldsymbol{e}_p$ subject to the forces generated on the hardware side. As stated above, these positions are fed back to the hardware to provide the operator with a realistic interaction feeling.

The human-drone interaction forces at the arm's tip $\boldsymbol{h}_h$, expressed in the world frame, are shown in Figure 10. After a few seconds from the beginning of the experiment, the operator grabs the hardware manipulator, and the interaction lasts for about 30 seconds until he successfully mounts the tool. At this point, the tool weight force of about $1.4N$ is the only one applied at the manipulator's end-effector. The admittance gains are diagonal matrices whose variation is shown in Figure 11. The arm starts with low gains to improve the comfort of the human operator and increase the system's safety. If a human contact is found (around 10 seconds), the virtual stiffness and the mass gains are increased to aid the tool's placement process. The increased gains violate the arm passivity, which can be guaranteed by employing an energy tank. The tank partially discharges as in Figure 12, consequently delaying the increase in the admittance gains. However, later in the experiment, the tank recharges thanks to the energy dissipated from the human interaction.

This second case study aims at installing a bird diverter on an aerial power line by impacting the line with sufficient force. The arm admittance controller gains are increased with respect to the human-UAM interaction phase to provide rigidity. Only the gains along the front direction are kept low to ease the diverter installation and attenuate the impact effects on the floating base.

In the beginning, the aerial manipulator is in free flight approaching the installation point. When this point is reached, the quadrotor is commanded to be still in position while the arm is positioned under the aerial cable and rapidly rises to hook the diverter. Because of the impact forces, shown in Figure 13, right after the diverter was hooked, the quadrotor undergoes a displacement of about $0.1m$ along the x-axis in Σ_w, which is recovered by the flight controller as shown in Figure 14.

Conclusion

As anticipated in Chapter 1, this chapter investigated the problem of an aerial manipulator interacting with the environment.

In its first part, the chapter has seen the implementation of the STSMC technique to an

aerial manipulator platform. The proposed control technique is a hybrid position tracking/force regulation approach, considering the entire manipulator system as a flying end-effector. Two case studies were performed within the Gazebo simulation environment to assess the STSMC: (i) tracking the desired trajectory and (ii) reaching a target position to apply lateral force for a long time. Both case studies showed successful tracking and force regulation results, yielding significantly small error norms, which proved the effectiveness of the proposed controller for omnidirectional aerial manipulator platforms. Future work will be devoted to implementing it on a real platform and testing the interaction with the environment with higher values of forces.

The second part of this chapter presented a general framework for HIL simulation of human-aerial manipulator collaboration. In the beginning, the simulated aerial manipulator and the hardware interface were presented. These two were then connected through a communication interface to implement the force and position feedback between the simulation and the environment. The overall architecture's effectiveness was evaluated in two case studies: (i) a collaborative task with a human operator and (ii) an autonomous bird diverter installation task. We demonstrated the possibility of performing human-aerial manipulator interaction during these experiments without endangering the operator. Additionally, the stability proof of the chosen architecture was proposed by employing both symbolic and numerical tools. Future work will further stress the HIL approach with other tasks and validate the possibility of using it to train human operators through a comprehensive human-subjects study. Besides, the proposed architecture can be tested with other hardware interfaces to stress its limits. Teleoperation may also be inserted with proper handling of the delay, thanks to the energy tank background.

Acknowledgment

The research leading to these results has been supported by both the AERIAL-CORE project, European Union's Horizon 2020 research and innovation programme, under Grant Agreement No 871479, and the AERO-TRAIN project, European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 953454. The authors are solely responsible for its content.

References

[1] J. Cacace, S. M. Orozco-Soto, A. Suarez, A. Caballero, M. Orsag, S. Bogdan, G. Vasiljevic, E. Ebeid, J. A. A. Rodriguez, and A. Ollero, "Safe local aerial manipulation for the installation of devices on power lines: Aerial-core first year results and designs," *Applied Sciences*, vol. 11, no. 13, p. 6220, 2021.

[2] F. Ruggiero, V. Lippiello, and A. Ollero, "Aerial manipulation: A literature review," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1957–1964, 2018.

[3] A. Ollero, M. Tognon, A. Suarez, D. Lee, and A. Franchi, "Past, present, and future of aerial robotics manipulators," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 626–645, 2022.

[4] D. Kotarski, M. Krznar, P. Piljek, and N. Simunic, "Experimental identification and characterization of multirotor UAV propulsion," *Journal of Physics: Conference Series*, vol. 870, p. 012003, 2017.

[5] A. Goel and A. Swarup, "Mimo uncertain nonlinear system control via adaptive high-order super twisting sliding mode and its application to robotic manipulator," *Journal of Control, Automation and Electrical Systems*, vol. 28, no. 1, pp. 36–49, 2017.

[6] Y. B. Shtessel, J. A. Moreno, F. Plestan, L. M. Fridman, and A. S. Poznyak, "Super-twisting adaptive sliding mode control: A lyapunov design," in *49th IEEE conference on decision and control (CDC)*. IEEE, 2010, pp. 5109–5113.

[7] W. Alqaisi, Y. Kali, J. Ghommam, M. Saad, and V. Nerguizian, "Position and attitude tracking of uncertain quadrotor unmanned aerial vehicles based on non-singular terminal super-twisting algorithm," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 234, no. 3, pp. 396–408, 2020.

[8] H. Ghadiri, M. Emami, and H. Khodadadi, "Adaptive super-twisting non-singular terminal sliding mode control for tracking of quadrotor with bounded disturbances," *Aerospace Science and Technology*, vol. 112, p. 106616, 2021.

[9] N. P. Nguyen, W. Kim, and J. Moon, "Observer-based super-twisting sliding mode control with fuzzy variable gains and its application to overactuated quadrotors," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 5993–5998.

[10] N. P. Nguyen, W. Kim, and J. Moon, "Super-twisting observer-based sliding mode control with fuzzy variable gains and its applications to fully-actuated hexarotors," *Journal of the Franklin Institute*, vol. 356, no. 8, pp. 4270–4303, 2019.

[11] R. Ji, J. Ma, S. S. Ge, and R. Ji, "Adaptive second-order sliding mode control for a tilting quadcopter with input saturations," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 3910–3915, 2020.

[12] S. Yi, K. Watanabe, and I. Nagai, "Anti-disturbance control of a quadrotor manipulator with tiltable rotors based on integral sliding mode control," *Artificial Life and Robotics*, pp. 1–10, 2021.

[13] S. Riache, M. Kidouche, and A. Rezoug, "Adaptive robust nonsingular terminal sliding mode design controller for quadrotor aerial manipulator," *TELKOMNIKA Telecommunication Comput. Electron. Control*, vol. 17, no. 3, pp. 1501–1512, 2019.

[14] C. Kuchwa-Dube and J. O. Pedro, "Altitude and attitude tracking of a quadrotor-based

aerial manipulator using super twisting sliding mode control," in *Proceedings of the 6th International Conference on Control, Mechatronics and Automation*, 2018, pp. 65–69.

[15] C. Kuchwa-Dube and J. O. Pedro, "Quadrotor-based aerial manipulator altitude and attitude tracking using adaptive super-twisting sliding mode control," in *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2019, pp. 144–151.

[16] V. Lippiello and F. Ruggiero, "Cartesian impedance control of a UAV with a robotic arm," in *10th International IFAC Symposium on Robot Control*, 2012, pp. 704–709.

[17] F. Ruggiero, J. Cacace, H. Sadeghian, and V. Lippiello, "Impedance control of VToL UAVs with a momentum-based external generalized forces estimator," in *2014 IEEE International Conference on Robotics and Automation*, 2014, pp. 2093–2099.

[18] D. Invernizzi, M. Giurato, P. Gattazzo, and M. Lovera, "Full pose tracking for a tilt-arm quadrotor uav," in *2018 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2018, pp. 159–164.

[19] M. Kamel, S. Verling, O. Elkhatib, C. Sprecher, P. Wulkop, Z. Taylor, R. Siegwart, and I. Gilitschenski, "The voliro omniorientational hexacopter: An agile and maneuverable tiltable-rotor aerial vehicle," *IEEE Robotics & Automation Magazine*, vol. 25, no. 4, pp. 34–44, 2018.

[20] M. Ryll, G. Muscio, F. Pierri, E. Cataldi, G. Antonelli, F. Caccavale, D. Bicego, and A. Franchi, "6d interaction control with aerial robots: The flying end-effector paradigm," *The International Journal of Robotics Research*, vol. 38, no. 9, pp. 1045–1062, 2019.

[21] J. Cacace, A. Finzi, V. Lippiello, M. Furci, N. Mimmo, and L. Marconi, "A control architecture for multiple drones operated via multimodal interaction in search rescue mission," in *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics*, 2016, pp. 233–239.

[22] A. C. S. Medeiros, P. Ratsamee, Y. Uranishi, T. Mashita, and H. Takemura, "Human-drone interaction: Using pointing gesture to define a target object," in *Human-Computer Interaction. Multimodal and Natural Interaction*, M. Kurosu, Ed. Springer International Publishing, Cham, 2020, pp. 688–705.

[23] D. Tezza, S. Garcia, T. Hossain, and M. Andujar, "Brain eRacing: An exploratory study on virtual brain-controlled drones," in *Virtual, Augmented and Mixed Reality. Applications and Case Studies (Lecture Notes in Computer Science)*, J. Chen and G. Fragomeni, Eds. Springer International Publishing, Cham, 2019, vol. 11575, pp. 150–162.

[24] J. Cacace, A. Finzi, and V. Lippiello, "A robust multimodal fusion framework for command interpretation in human-robot cooperation," in *2017 26th IEEE International Symposium on Robot and Human Interactive Communication*, 2017, pp. 372–377.

[25] L. E. Jane, L. E. Ilene, J. A. Landay, and J. R. Cauchard, "Drone & Wo: Cultural influences on human-drone interaction techniques," in *2017 CHI Conference on Human Factors in*

Computing Systems, 2017, pp. 6794—-6799.

[26] D. Tezza and M. Andujar, "The state-of-the-art of human-drone interaction: A survey," *IEEE Access*, vol. 7, pp. 167 438–167 454, 2019.

[27] M. Selvaggio, M. Cognetti, S. Nikolaidis, S. Ivaldi, and B. Siciliano, "Autonomy in physical human-robot interaction: A brief survey," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7989–7996, 2021.

[28] P. Abtahi, D. Y. Zhao, J. L. E., and J. A. Landay, "Drone near me: Exploring touch-based human-drone interaction," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, 2017.

[29] M. Lieser, U. Schwanecke, and J. Berdux, "Tactile human-quadrotor interaction: Metro-drone," in *Proceedings of the Fifteenth International Conference on Tangible, Embedded, and Embodied Interaction*, 2021, pp. 1–6.

[30] S. Rajappa, H. Bülthoff, and P. Stegagno, "Design and implementation of a novel architecture for physical human-UAV interaction," *The International Journal of Robotics Research*, vol. 36, no. 5–7, pp. 800–819, 2017.

[31] F. Augugliaro and R. D'Andrea, "Admittance control for physical human-quadcopter interaction," in *2013 European Control Conference*, 2013, pp. 1805–1810.

[32] M. Tognon, R. Alami, and B. Siciliano, "Physical human-robot interaction with a tethered aerial vehicle: Application to a force-based human guiding problem," *IEEE Transactions on Robotics*, pp. 1–12, 2021.

[33] E. Cuniato, J. Cacace, M. Selvaggio, F. Ruggiero, and V. Lippiello, "A hardware-in-the-loop simulator for physical human-aerial manipulator cooperation," in *20th International Conference on Advanced Robotics*, 2021.

[34] F. Furrer, M. Burri, and R. Achtelik, M. aand Sigwart, "RotorS—A modular Gazebo MAV simulator framework," in *Robot Operating System (ROS). Studies in Computational Intelligence*, K. A. (eds), Ed. Springer, Cham, 2016, vol. 625.

[35] T. Madani and A. Benallegue, "Backstepping control for a quadrotor helicopter," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 3255—-3260.

[36] R. Murray, Z. Li, and S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 2017.

[37] T. Lee, M. Leok, and N. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," in *9th IEEE Conference on Decision and Control*, 2010, pp. 5420–5425.

[38] J. Slotine and W. Li, "On the adaptive control of robot manipulators," *International Jounal of Robotics Research*, vol. 6, no. 3, pp. 49—-59, 1987.

[39] H. Berghuis and H. Nijmeijer, "A passivity approach to controller–observer design for robots," *IEEE Transactions on Robotics and Automation*, vol. 9, pp. 740—-754, 1993.

[40] F. Ferraguti, C. Secchi, and C. Fantuzzi, "A tank-based approach to impedance control with variable stiffness," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 4948–4953.

[41] M. Selvaggio, P. Robuffo Giordano, F. Ficuciello, and B. Siciliano, "Passive task-prioritized shared-control teleoperation with haptic guidance," in *2019 IEEE International Conference on Robotics and Automation*, 2019, pp. 430–436.

[42] F. Ruggiero, M. Trujillo, R. Cano, H. Ascorbe, A. Viguria, C. Peréz, V. Lippiello, A. Ollero, and B. Siciliano, "A multilayer control for multirotor uavs equipped with a servo robot arm," in *2015 IEEE International Conference on Robotics and Automation*, 2015, pp. 4014–4020.

[43] H. K. Khalil, *Nonlinear systems; 3rd ed.* Upper Saddle River, NJ: Prentice-Hall, 2002, the book can be consulted by contacting: PH-AID: Wallet, Lionel. [Online]. Available: https://cds.cern.ch/record/1173048

[44] W. R. Inc., "Mathematica, Version 12.3.1," champaign, IL, 2021. [Online]. Available: https://www.wolfram.com/mathematica

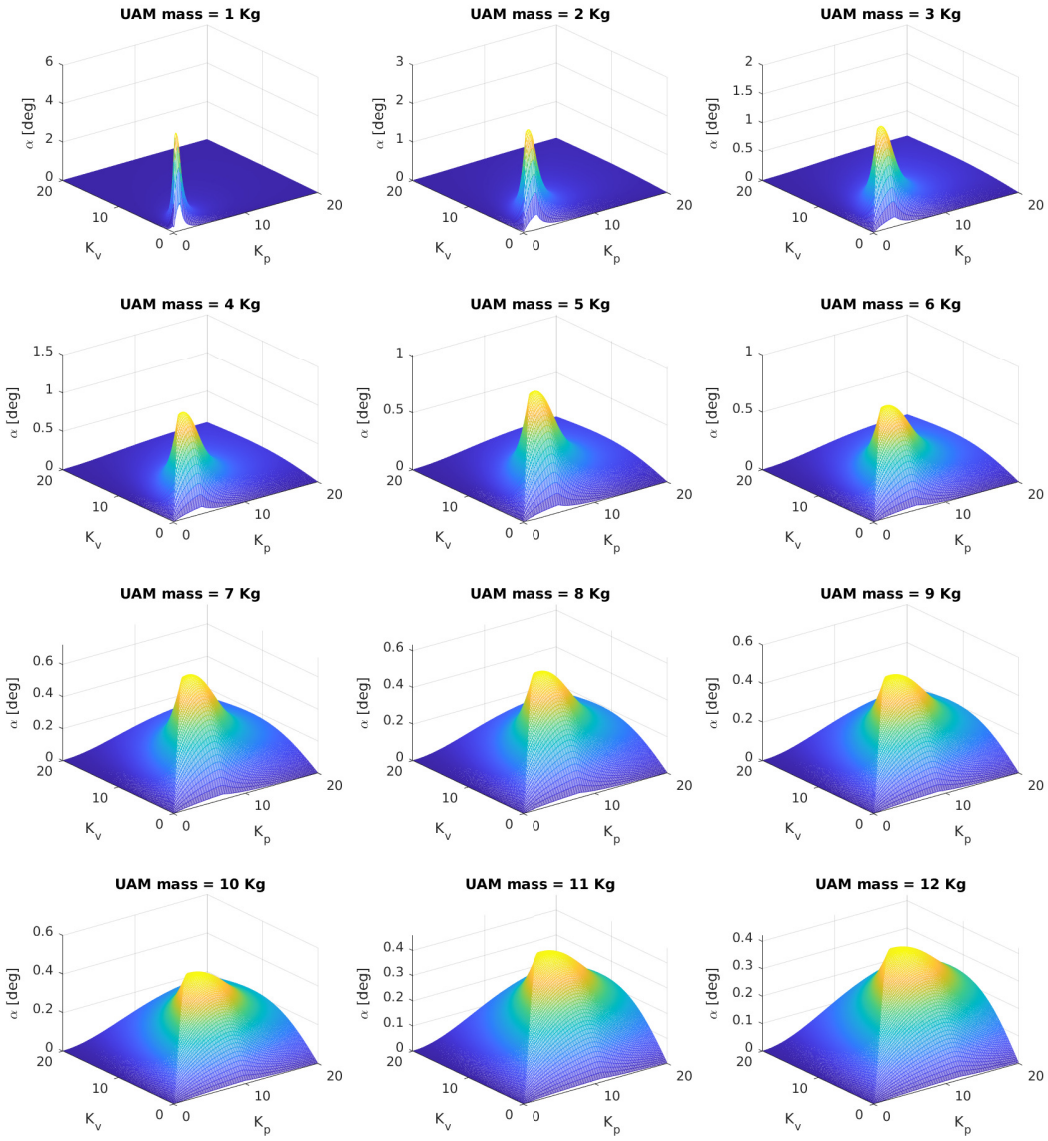

Figure 8: Limits on the initial attitude tracking error for the existence of bounds on the UAM state. The limits are function of the UAM mass as well as the UAV controller gains.

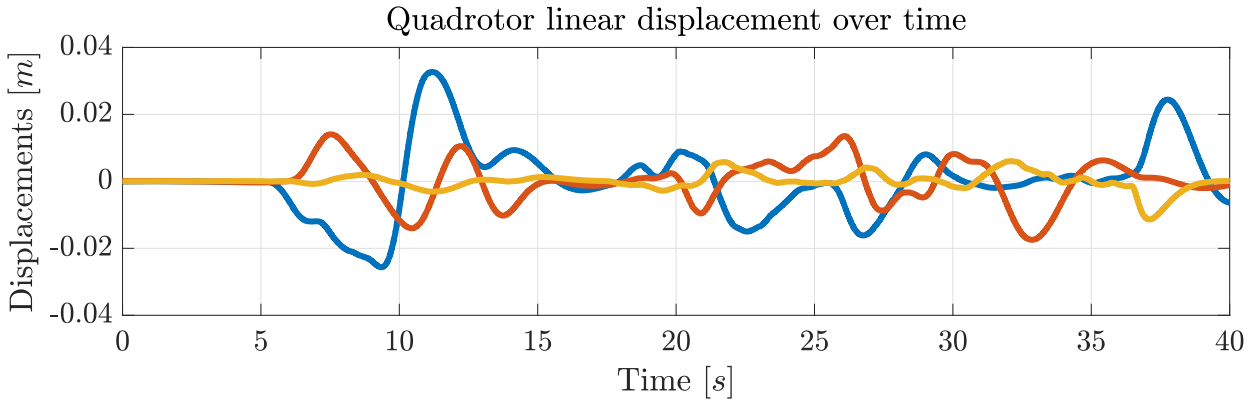

Figure 9: Aerial base position displacement $\boldsymbol{e}_p$ along axes x (blue), y (red) and z (orange) during human interaction experiment.

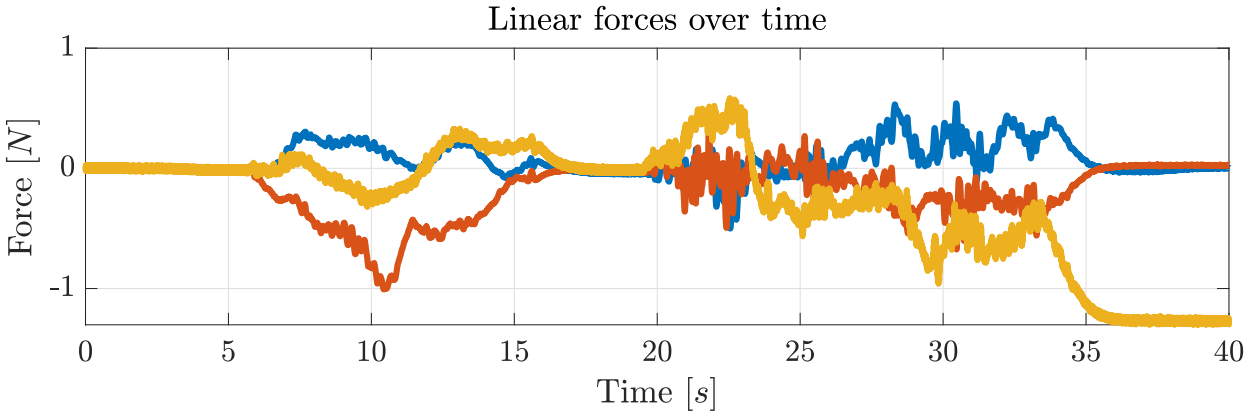

Figure 10: Human-hardware contact forces $\boldsymbol{h}_{e'}^{b'}$ along axes x (blue), y (red) and z (orange) during human interaction experiment.

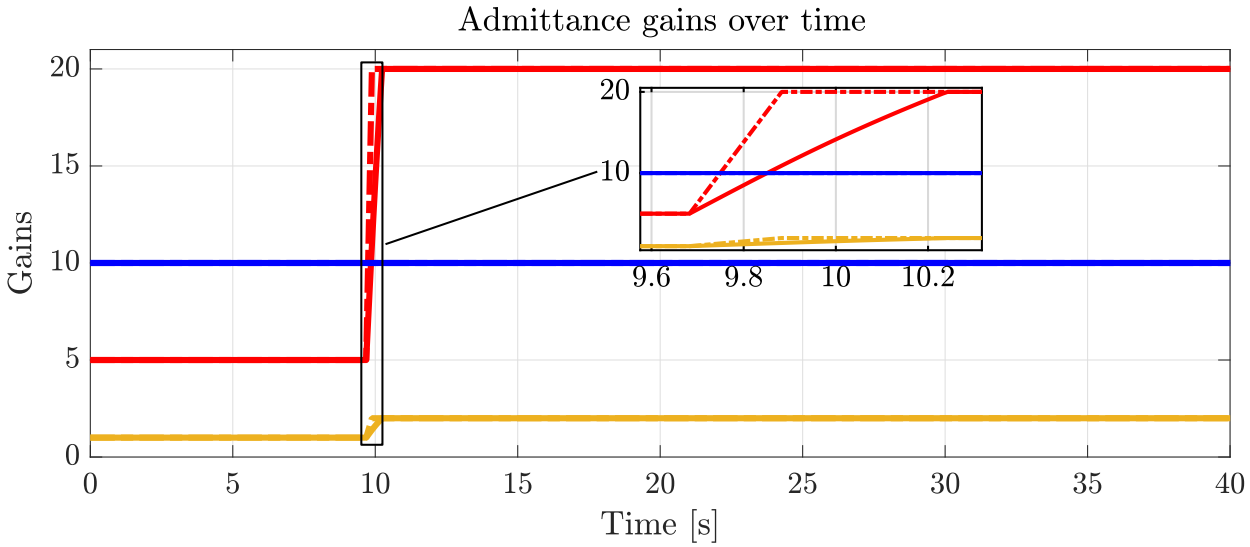

Figure 11: Quadrotor arm admittance gains $\mathbf{K}_P$ (red), $\mathbf{K}_D$ (blue) and $\mathbf{M}_d$ (orange) during human interaction experiment: desired (dashed) and actual profiles (solid).

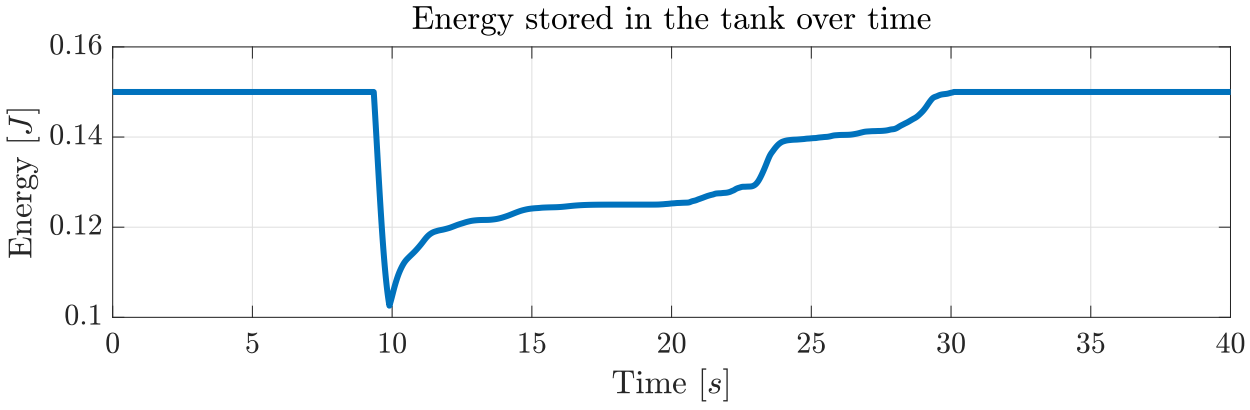

Figure 12: Energy $\mathcal{T}$ inside the tank during human interaction experiment.

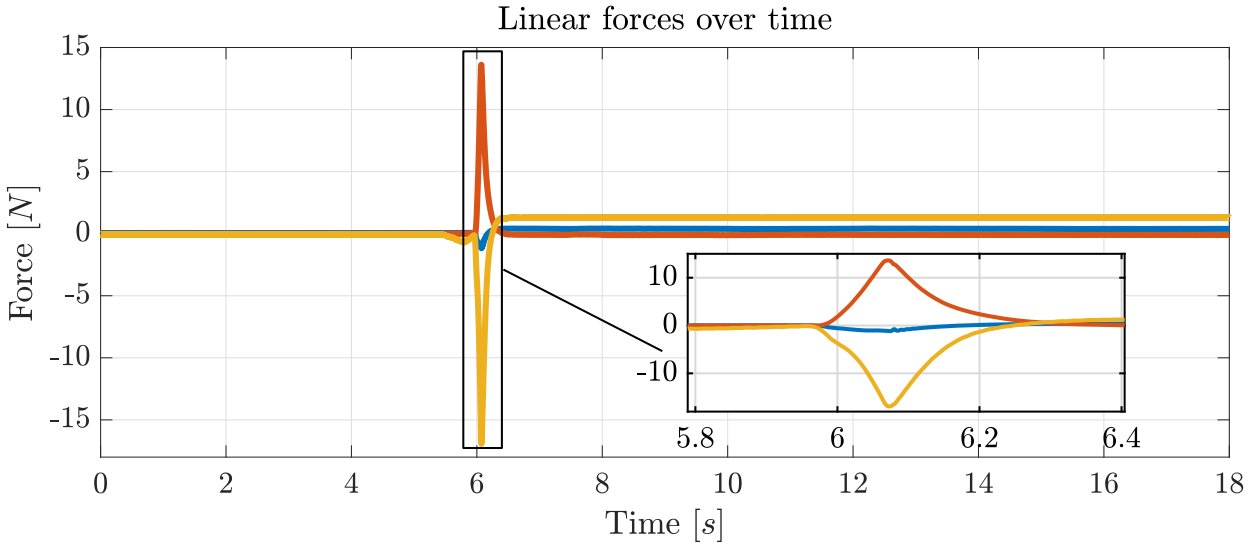

Figure 13: Diverter installation impact forces $\boldsymbol{h}_{e'}^{b'}$ along axes x (blue), y (red) and z (orange) during the bird diverter installation experiment.

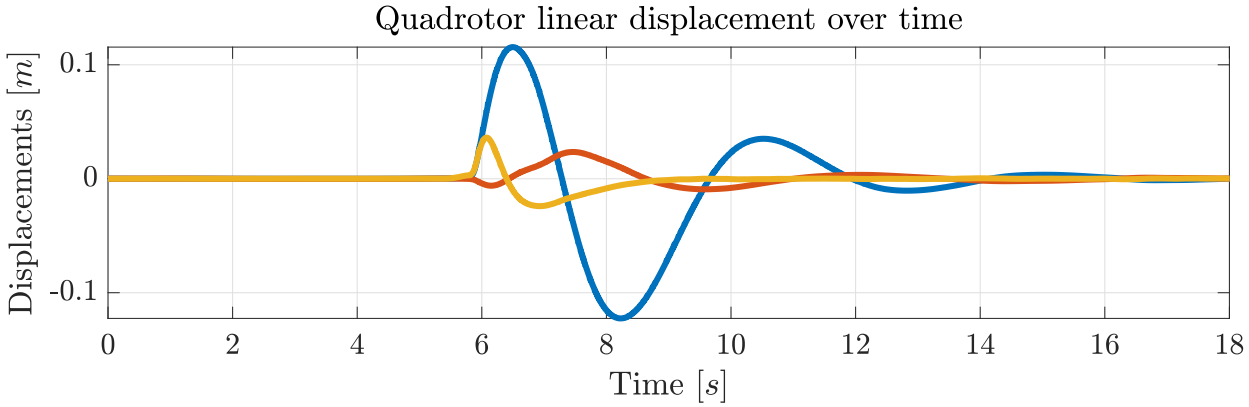

Figure 14: Aerial base position displacement $\boldsymbol{e}_p$ in Σ_w along axes x (blue), y (red) and z (orange) during the bird diverter installation experiment.