

Full-Pose Tracking Control for Aerial Robotic Systems with Laterally-Bounded Input Force

Antonio Franchi, *Senior Member, IEEE*, Ruggero Carli, *Member, IEEE*, Davide Bicego, and Markus Ryll, *Member, IEEE*

Abstract—A class of abstract aerial robotic systems is introduced, the Laterally Bounded Force (LBF) vehicles, in which most of the control authority is expressed along a principal thrust direction, while along the lateral directions a (smaller and possibly null) force may be exploited to achieve full-pose tracking. This class approximates platforms endowed with non-collinear rotors that can modify the orientation of the total thrust in body frame. The proposed SE(3)-based control strategy achieves, if made possible by the force constraints, the independent tracking of position-plus-orientation trajectories. The method, which is proven using a Lyapunov technique, deals seamlessly with both *under-* and *fully-actuated* platforms, and guarantees at least the position tracking in the case of an unfeasible full-pose reference trajectory. Several experimental tests are presented, that clearly show the approach practicability and the sharp improvement with respect to state-of-the-art.

I. INTRODUCTION

In the last years we have seen advancements in control, perception and actuation allowing unmanned aerial vehicles (UAVs) to perform very agile maneuvers [1], as well as complex missions solely or in swarms and heterogenous groups. Originally performed missions have been contactless as, e.g., environmental monitoring and search and rescue. Nowadays UAVs are as well utilized as aerial *robots* to perform direct physical interaction – UAVs grasp, transport, and manipulate our environment, as shown by several research projects on aerial manipulation¹, and can also physically interact through cables [2], [3] in order to perform unprecedented maneuvers [4].

Standard multi-rotors (quadrotors, hexarotors, etc.) have collinear propellers generating forces that are all aligned to one direction in body frame, which makes them under-actuated systems: the total control force can be produced only along that direction. In order to follow a generic position trajectory the total force direction in world frame is changed by rotating the whole vehicle. Maneuvers in which rotation and translation are completely independent are precluded to such platforms, which constitutes a serious problem in the case that, e.g., the platform has to move through a hostile and cluttered

ambient or resist a wind gust while keeping a desired attitude. Such an underactuation even deteriorates the potentiality to interact with the environment by rapidly exerting forces in an arbitrarily-chosen direction of the space while keeping a pre-specified orientation.

The major solution to overcome the aforementioned issues has been to mount rotors in a tilted way such that the thrusts of the propellers are not collinear anymore. In this way, the direction of the total force can be changed by selecting the intensity of the force produced by each propeller, without the need of reorienting the whole vehicle. If the propellers are at least six, and the tilting directions do not generate a singular configuration, then direction and intensity of both the instantaneous total control moment and total control force are controllable at will. This idea has been implemented in several real platforms and is becoming more and more popular in the aerial vehicles and robotics communities. For example, results achieved in [5] show an improvement in resisting an opposing wrench, while the works done in [6]–[8] show that such idea allows to decouple the tracking of a desired position and orientation.

These new kind of *multi-directional thrust* platforms (sometimes referred to as *fully-actuated*) call for new methods to control them efficiently and to reliably cope with the added complexity of the platforms and of the larger set of tasks in which they can be employed, when compared to standard collinear multi-rotors. To fill this gap, in this paper we propose a novel method for controlling such platforms while taking into account the most limiting input bounds they have to cope with, i.e., lateral input force. The proposed controller ensures, in nominal conditions, the tracking of a full-6D pose reference trajectory (position plus orientation). If the reference orientation and the force needed to track the position trajectory do not comply with the platform constraints, the proposed strategy gives priority to the tracking of the positional part while also tracking the feasible orientation that is the closest to the reference one. This choice is supported by, e.g., the fact that in typical applications a wrong position tracking is more likely to lead to an obstacle crash than a non-perfect orientation tracking.

In order to attain generality, the proposed method is designed to work with a large variety of different platforms which include not only the fully-actuated but also the under-actuated (collinear propeller) case. In this way we both maximize the breadth of the impact of the proposed methodology to new full-actuated platforms and we also maintain back-compatibility with standard platforms. This last feature makes

Antonio Franchi, Davide Bicego and Markus Ryll are with LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France, antonio.franchi@laas.fr, dbicego@laas.fr, mryll@laas.fr

Ruggero Carli is with Department of Information Engineering, University of Padova, Padova, Italy, carlirug@dei.unipd.it

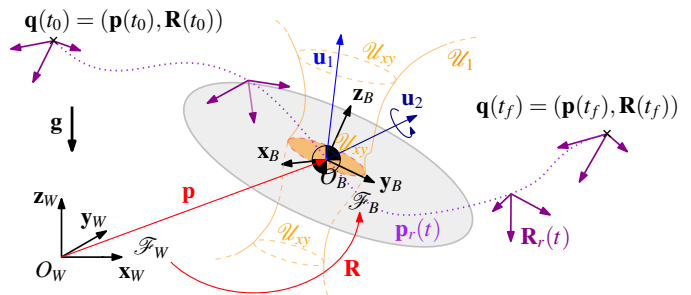
This work has been funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 644271 AEROARMS.

¹ARCAS, EU Coll. Proj. ICT-287617, www.arcas-project.eu;
AeRoArms, EU Coll. Proj. ICT-644271, www.aerarms-project.eu;
AEROworks, EU Coll. Proj. ICT-644128, www.aeroworks2020.eu.

the proposed method also suitable to control vectored-thrust vehicles that can transit from an under-actuated to a fully-actuated configuration while flying – as, e.g., the one presented in [9]. The proposed method is universal and does *not* need any switching between two different controllers for each configuration.

Motivations and State of the Art: To optimize the use of available energy, common multi-rotor platforms are designed with all collinear rotors, at the expense of underactuation. For these platforms several controllers have been proposed like, e.g., the ones based on the dynamic feedback linearization [11], cascaded/backstepping ones [12], [13], and geometric ones on $SE(3)$ [14], [15] (see [16] for a review).

An underactuated LBF platform is not able to track a generic full-pose trajectory, i.e., with independent position and orientation in $SE(3)$. The rotation about any axis that is orthogonal to the principal fixed total thrust direction must follow the evolution over time of the position trajectory, according to the well-known differential flatness property [1], [11]. Therefore an underactuated multi-rotor aerial platform can only track a 4D-pose trajectory (i.e., position plus the rotation about the principal direction). On the contrary, a ‘fully-actuated’ LBF platform can exert some force in the lateral direction thus allowing the tracking of a generic full-pose (6D) trajectory. However, due to the bounded thrust along the lateral directions, it is not possible to track *any* full-pose trajectory. The larger the bounds the higher the ability of the platform to track any trajectory, the lower the bounds the more the platform resembles an underactuated multi-rotor and thus it becomes almost unable to track a full-pose trajectory but only a 4D-pose one.



the nonlinear dynamical effects and trying to zero the position and orientation errors. Then the thrust inputs for each propeller are computed from the control wrench by simply inverting the control allocation matrix. This method has been first proposed in [6] and then used also in [7] (with pseudo-inversion, in place of inversion, to allocate the input redundancy) and in [8]. The limitation of this control approach is to neglect input saturation, which may easily lead to an unstable behavior if, e.g., the full-pose trajectory to be followed is not input-feasible. A preliminary attempt to solve these issue in a special case has been done in [18], where a weighted pseudoinversion is used to allocate the 6 inputs of a coplanar hexarotor in order to obtain a lower dimensional task (4D) output (the total thrust and the 3 moment components). Another control approach is presented in [17], which is however specific to that octorotor platform, it does not consider input bounds either, and is based on a particular Euler angle representation. Finally, [19] presents an approach that also uses Euler and is based on the reference governor. The method is tested only in simulation and for constant position and attitude references.

The remaining part of the paper is structured as follows. Section II presents the generic model. The proposed control method is introduced in Sec. III. Section IV shows the full computation of the generic controller in a meaningful case, while results of several experiments are shown in Sec. V. Finally we conclude the paper and give an outline of further possible extensions in Sec. VI.

An LBF platform is a rigid body to which gravity and control generalized forces are applied. For the sake of compactness, all the main symbols used in the paper are summarized in Table I and Fig. 1.

TABLE I
MAIN SYMBOLS USED IN THE PAPER

Definition	Symbol
World Inertial Frame: $\mathcal{F}_W = O_W, \{\mathbf{x}_W, \mathbf{y}_W, \mathbf{z}_W\}$	\mathcal{F}_W
Attached Body Frame: $\mathcal{F}_B = O_B, \{\mathbf{x}_B, \mathbf{y}_B, \mathbf{z}_B\}$	\mathcal{F}_B
Position of O_B in \mathcal{F}_W , coincident with the Center of Mass (CoM)	\mathbf{p}
Rotation matrix mapping coordinates in \mathcal{F}_B to coordinates in \mathcal{F}_W	\mathbf{R}
Configuration of the vehicle	$\mathbf{q} = (\mathbf{p}, \mathbf{R})$
Angular velocity of \mathcal{F}_B w.r.t \mathcal{F}_W expr. in \mathcal{F}_B	$\boldsymbol{\omega}$
Mass of the vehicle	m
Vehicle's Inertia matrix w.r.t to O_B expressed in \mathcal{F}_B	\mathbf{J}
Control force applied at the CoM expressed in \mathcal{F}_B	\mathbf{u}_1
Control moment applied at the CoM expressed in \mathcal{F}_B	\mathbf{u}_2
Feasible set of the control force \mathbf{u}_1	\mathcal{U}_1
Feasible set of the projection of \mathbf{u}_1 on the xy plane in \mathcal{F}_B	\mathcal{U}_{xy}
i -th vector of the canonical basis of \mathbb{R}_3 with $i = 1, 2, 3$	\mathbf{e}_i
Reference position for \mathbf{p} at time t	$\mathbf{p}_r(t)$
Reference rotation matrix for \mathbf{R} at time t	$\mathbf{R}_r(t)$
Reference control force to be applied at O_B	$\mathbf{f}_r(t)$ at time t
Set of orientations in $SO(3)$ that allow the application of $\mathbf{f}_r(t)$	$\mathcal{R}(\mathbf{f}_r)$
Subset of $\mathcal{R}(\mathbf{f}_r)$ that minimizes a certain cost w.r.t. \mathbf{R}_r	$\mathcal{R}(\mathbf{f}_r, \mathbf{R}_r)$
Desired rotation matrix in $\mathcal{R}(\mathbf{f}_r, \mathbf{R}_r)$	\mathbf{R}_d

to the vehicle expressed in \mathcal{F}_B . The following constraint applies

$$[u_1 \ u_2]^T \in \mathcal{U}_{xy} \subset \mathbb{R}^2, \quad (1)$$

where the *laterally bounding* set \mathcal{U}_{xy} is a set that contains the origin. We define $\mathcal{U}_1 = \{\mathbf{u}_1 \in \mathbb{R}^3 \mid [u_1 \ u_2]^T \in \mathcal{U}_{xy}, u_3 \geq 0\}$. Note that \mathcal{U}_{xy} can be constant or even be changing depending of u_3 , as shown in Figure 1. The dynamics of the aerial platform is then

$$m\ddot{\mathbf{p}} = -m\mathbf{g}\mathbf{e}_3 + \mathbf{R}\mathbf{u}_1, \quad \mathbf{J}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} + \mathbf{u}_2. \quad (2)$$

Some particularly relevant cases of LBF follow.

Case 1 (Underactuated aerial vehicle) When $\mathcal{U}_{xy} = \{\mathbf{0}\}$ the total force is always oriented as $\mathbf{R}\mathbf{e}_3$ and model (2) becomes the underactuated quadrotor model considered in [11], [13], [15]. *Case 2*: (Conic LBF) When $\mathcal{U}_{xy} = \{[u_1 \ u_2]^T \in \mathbb{R}^2 \mid u_1^2 + u_2^2 \leq (\tan \alpha)^2 u_3^2\}$, model (2) approximates the case of hexarotors with tilted propellers [5], [6], [9], for which the set of allowable \mathcal{U}_1 forces has the pseudo-inverted-pyramidal shape. The quantity α is a parameter that represents the tilting angle of the propellers (hexarotor).

Case 3: (Cylindric LBF) When

$$\mathcal{U}_{xy} = \{[u_1 \ u_2]^T \in \mathbb{R}^2 \mid u_1^2 + u_2^2 \leq r_{xy}^2\}, \quad (3)$$

model (2) approximates the case of an multirotor with n_m main propellers pointing up and n_s secondary less powerful propellers tilted 90 degrees w.r.t. the main ones, like the one presented in [17], for which the set of allowable \mathcal{U}_1 forces can be approximated by a pseudo-cylindric shape

III. FULL-POSE GEOMETRIC CONTROL ON SE(3)

Let be given a full-pose trajectory $\mathbf{q}_r(t) = (\mathbf{p}_r(t), \mathbf{R}_r(t)) : [t_0, t_f] \rightarrow SE(3)$, where $\mathbf{p}_r(t) \in \mathbb{R}^3$ is the reference position trajectory and $\mathbf{R}_r(t) \in SO(3)$ is the reference attitude trajectory. Inverting (2), the nominal inputs to track $\mathbf{q}_r(t)$ are obtained as $\mathbf{u}_1^r = \mathbf{R}_r^T (m\mathbf{g}\mathbf{e}_3 + m\ddot{\mathbf{p}}_r)$ and $\mathbf{u}_2^r = \boldsymbol{\omega}_r \times \mathbf{J}\boldsymbol{\omega}_r + \mathbf{J}\dot{\boldsymbol{\omega}}_r$, where $\boldsymbol{\omega}_r$ is defined by $[\boldsymbol{\omega}_r]_\times = \mathbf{R}_r^T \dot{\mathbf{R}}_r$.

Definition 1. $\mathbf{q}_r(t)$ is *feasible* if $\mathbf{u}_1^r(t) \in \mathcal{U}_1 \ \forall t \in [t_0, t_f]$.

Exact full-pose (6D) tracking is possible only if $\mathbf{q}_r(t)$ is feasible. However in real world it is not granted that $\mathbf{q}_r(t)$ will be such for the particular LBF platform in use. For this reason, we propose a controller that works (in the sense that the tracking of $\mathbf{p}_r(t)$ is still guaranteed and no singularity appears) even if $\mathbf{q}_r(t)$ is not feasible. Consider the position error $\mathbf{e}_p = \mathbf{p} - \mathbf{p}_r$, velocity error $\mathbf{e}_v = \dot{\mathbf{p}} - \dot{\mathbf{p}}_r$, and two positive definite gain matrices \mathbf{K}_p and \mathbf{K}_v . Then consider $\mathbf{f}_r = m\ddot{\mathbf{p}}_r + m\mathbf{g}\mathbf{e}_3 - \mathbf{K}_p\mathbf{e}_p - \mathbf{K}_v\mathbf{e}_v$, representing the reference total control force that ideally one would like to apply to the aerial vehicle CoM if the system would be completely fully actuated, i.e., if $\mathcal{U}_1 = \mathbb{R}^3$.

The set of orientations that allow to apply \mathbf{f}_r to the CoM of the LBF aerial vehicle is defined as $\mathcal{R}(\mathbf{f}_r) = \{\mathbf{R} \in SO(3) \mid \mathbf{R}^T \mathbf{f}_r \in \mathcal{U}_1\}$. For an underactuated collinear multi-rotor system the set $\mathcal{R}(\mathbf{f}_r)$ is formed by any \mathbf{R} such that $\mathbf{R}\mathbf{e}_3$ and \mathbf{f}_r are parallel, i.e., $\mathbf{R}\mathbf{e}_3 \times \mathbf{f}_r = \mathbf{0}$. For a generic LBF aerial vehicle the set $\mathcal{R}(\mathbf{f}_r)$ may contain also \mathbf{R} 's for which $\mathbf{R}\mathbf{e}_3 \times \mathbf{f}_r \neq \mathbf{0}$. Therefore we have the following.

Proposition 1. The set $\mathcal{R}(\mathbf{f}_r)$ is always nonempty $\forall \mathbf{f}_r \in \mathbb{R}^3$.

Proof. If $\mathbf{f}_r \neq \mathbf{0}$ then, by definition of $\mathcal{R}(\mathbf{f}_r)$ and \mathcal{U}_1 , $\mathcal{R}(\mathbf{f}_r) \supset \{\mathbf{R} \in SO(3) \mid \mathbf{R}\mathbf{e}_3 = \frac{\mathbf{f}_r}{\|\mathbf{f}_r\|}\} \neq \emptyset$. If $\mathbf{f}_r = \mathbf{0}$ then $\mathcal{R}(\mathbf{f}_r) = SO(3)$. \square

The proposed controller exploits a cascaded structure² by choosing, at each time t , a desired orientation $\mathbf{R}_d \in SO(3)$ that belongs to $\mathcal{R}(\mathbf{f}_r)$ and also minimizes a given cost function w.r.t. \mathbf{R}_r . Then one can use the fully actuated rotational dynamics to track \mathbf{R}_d and, in turn, track the reference position \mathbf{p}_r . If \mathbf{q}_r is feasible then \mathbf{R}_d will exponentially converge to \mathbf{R}_r . Otherwise, only the best feasible orientation will be obtained. Therefore the controller implicitly prioritizes the position trajectory tracking against the orientation one.

Define $\overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r) \subset \mathcal{R}(\mathbf{f}_r)$ as the set of rotation matrices that solve $\min_{\mathbf{R}' \in \mathcal{R}(\mathbf{f}_r)} J(\mathbf{R}_r, \mathbf{R}')$, where $J : SO(3) \times SO(3) \rightarrow \mathbb{R}_{\geq 0}$ is an arbitrarily chosen cost function that represents the degree of similarity between \mathbf{R}_r and \mathbf{R}' one is interested in. The elements in $\overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r)$ represent orientations of the LBF that allow to apply \mathbf{f}_r and minimize the function J w.r.t. \mathbf{R}_r .

Consider that, at each time t a desired orientation $\mathbf{R}_d \in \overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r)$ is chosen. Furthermore, whenever $\mathbf{R}_r \in \overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r)$ then \mathbf{R}_d must be chosen equal to \mathbf{R}_r . Then define the rotation error $\mathbf{e}_R = \frac{1}{2}(\mathbf{R}_d^T \mathbf{R} - \mathbf{R}^T \mathbf{R}_d)^\vee$, and the angular velocity error $\mathbf{e}_\omega = \boldsymbol{\omega} - \mathbf{R}^T \mathbf{R}_d \boldsymbol{\omega}_d$ where \bullet^\vee is the inverse map of $[\bullet]_\times$, and $\boldsymbol{\omega}_d$ is the angular velocity associated to \mathbf{R}_d . Consider then the following control law

$$\mathbf{u}_1 = \text{sat}_{\mathcal{U}_{xy}}((\mathbf{f}_r^T \mathbf{R}_d \mathbf{e}_1)(\mathbf{f}_r^T \mathbf{R}_d \mathbf{e}_2) + (\mathbf{f}_r^T \mathbf{R}_d \mathbf{e}_3)\mathbf{e}_3) \quad (4)$$

$$\mathbf{u}_2 = \boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} - \mathbf{K}_R \mathbf{e}_R - \mathbf{K}_\omega \mathbf{e}_\omega - \mathbf{J}([\boldsymbol{\omega}]_\times \mathbf{R}_d^T \mathbf{R}_d \boldsymbol{\omega}_d - \mathbf{R}^T \mathbf{R}_d \dot{\boldsymbol{\omega}}_d) \quad (5)$$

where $\text{sat}_{\mathcal{U}_{xy}}(\mathbf{x})$ is a vector in \mathcal{U}_{xy} with the same direction of \mathbf{x} , that minimizes the distance from \mathbf{x} . $\mathbf{K}_R = k_R \mathbf{I}$ and $\mathbf{K}_\omega = k_\omega \mathbf{I}$ are the gain matrices with $k_R > 0$ and $k_\omega > 0$.

²Notice that even if a cascaded structure is used, there is no time-scale separation assumption in the proposed controller.

Notice that only in the very special case in which $\mathcal{U}_{xy} = \{\mathbf{0}\}$ the control in (4)–(5) resembles the one in [15] while its form is in general very different from it. Another difference with [15] is that the computation of \mathbf{R}_d in [15] is done from the position trajectory using the differential flatness property while here is computed ensuring input feasibility and minimizing J w.r.t. \mathbf{R}_r .

In order to prove the convergence properties of the proposed controller let us consider the following error function between two rotation matrixes \mathbf{R}_1 and \mathbf{R}_2 to be $d(\mathbf{R}_1, \mathbf{R}_2) = \frac{1}{2} \text{tr}(\mathbf{I} - \mathbf{R}_2^T \mathbf{R}_1)$.

Theorem 1. Assume that $\mathbf{R}_d(t) \in \mathcal{R}(\mathbf{f}_r(t))$ for any t and that $\boldsymbol{\omega}_d(t)$ and $\dot{\boldsymbol{\omega}}_d(t)$ are well defined for any t . Consider the control \mathbf{u}_1 and \mathbf{u}_2 defined at (4) and (5).

Assume that the initial condition satisfies

$$d(\mathbf{R}(0), \mathbf{R}_d(0)) < 2, \quad (6)$$

$$\|\mathbf{e}_\omega(0)\|^2 < \frac{2}{\lambda_{\min}(\mathbf{J})} k_R (1 - d(\mathbf{R}(0), \mathbf{R}_d(0))) \quad (7)$$

Then, the zero equilibrium of the tracking errors \mathbf{e}_R , \mathbf{e}_ω , \mathbf{e}_p and \mathbf{e}_v is exponentially stable. The region of attraction is characterized by (6) and (7).

Proof. The proof is divided into two parts. We first show that, if the $\mathbf{R}(0)$ and $\mathbf{e}_\omega(0)$ satisfy, respectively, (6) and (7), then $\mathbf{R}(t)$ converges exponentially to $\mathbf{R}_d(t)$, in the sense that the function $d(\mathbf{R}(t), \mathbf{R}_d(t))$ goes exponentially to zero. Secondly, we characterize the translational error dynamics and, based on the fact that $\mathbf{R}(t)$ converges exponentially to $\mathbf{R}_d(t)$, we show that also \mathbf{e}_p and \mathbf{e}_v goes exponentially to zero. The time derivative of \mathbf{e}_ω is

$$\mathbf{J}\dot{\mathbf{e}}_\omega = \mathbf{J}\dot{\boldsymbol{\omega}} + \mathbf{J}([\boldsymbol{\omega}]_\times \mathbf{R}^T \mathbf{R}_d \boldsymbol{\omega}_d - \mathbf{R}^T \mathbf{R}_d \dot{\boldsymbol{\omega}}_d). \quad (8)$$

Plugging the rotational part of (2) into (8) and substituting \mathbf{u}_2 from (5), we get $\mathbf{J}\dot{\mathbf{e}}_\omega = -k_R \mathbf{e}_R - k_\omega \mathbf{e}_\omega$. In [15], it is shown, by exhibiting a suitable Lyapunov function, that, under conditions in (6) and in (7), the zero equilibrium of the attitude tracking error \mathbf{e}_R , \mathbf{e}_ω is exponentially stable and that there exist two positive constants α, β such that

$$d(\mathbf{R}(t), \mathbf{R}_d(t)) < \alpha e^{-\beta t} d(\mathbf{R}(0), \mathbf{R}_d(0)). \quad (9)$$

We determine now the error dynamics of the translational dynamics. Substituting \mathbf{u}_1 from (4) in the translational part of (2) we obtain

$$m\ddot{\mathbf{p}} = -mg\mathbf{e}_3 + \mathbf{f}_r + \mathbf{R}\mathbf{u}_1 - \mathbf{f}_r = m\ddot{\mathbf{p}}_r - \mathbf{K}_p \mathbf{e}_p - \mathbf{K}_v \mathbf{e}_v + \boldsymbol{\gamma},$$

where $\boldsymbol{\gamma} = \mathbf{R}\mathbf{u}_1 - \mathbf{f}_r$. It easily follows that

$$m\dot{\mathbf{e}}_v = -\mathbf{K}_p \mathbf{e}_p - \mathbf{K}_v \mathbf{e}_v + \boldsymbol{\gamma}. \quad (10)$$

Consider (9) and observe that, since $\mathbf{R}_d \in \mathcal{R}(\mathbf{f}_r)$ for any t , we have that there exist two positive constants C, ρ such that

$$\|\boldsymbol{\gamma}(t)\| \leq C e^{-\rho t} \|\boldsymbol{\gamma}(0)\|.$$

Let $\mathbf{x} = [\mathbf{e}_v \ \mathbf{e}_p]^T$ then, (10) can be written in vector form as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\boldsymbol{\gamma} \quad (11)$$

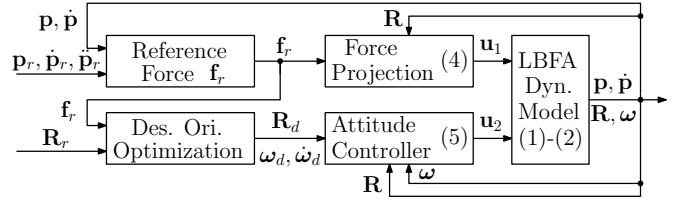


Fig. 2. Block diagram of the proposed geometric controller with the references to the corresponding equations in the text.

where

$$\mathbf{A} = \frac{1}{m} \begin{bmatrix} -\mathbf{K}_v & -\mathbf{K}_p \\ \mathbf{I} & \mathbf{0} \end{bmatrix}, \quad \mathbf{B} = \frac{1}{m} \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix}.$$

Since $\mathbf{K}_v, \mathbf{K}_p$ are both positive definite matrices, we have that \mathbf{A} is a Hurwitz matrix.

Observe that (11) is the cascade of a linear stable system and an exponential stable signal. Then, the statement of the Theorem follows from a technical refinement of Lemma 4.7 in [20], i.e., the cascade of a asymptotically stable linear system and of an exponentially stable system is exponentially stable. \square

A block diagram that shows the main subsystems of the proposed control architecture is provided in Fig. 2.

Theorem 1 ensures, under mild conditions, the exponential stability of \mathbf{e}_p , \mathbf{e}_v , \mathbf{e}_R , and \mathbf{e}_ω . Notice that this results holds regardless of the feasibility of \mathbf{q}_r . If \mathbf{q}_r is also feasible then exponential tracking of \mathbf{q}_r by \mathbf{q} is also guaranteed. In order to formally state this fact let us define the errors $\mathbf{e}_{R_r} = \frac{1}{2}(\mathbf{R}_r^T \mathbf{R}_d - \mathbf{R}_d^T \mathbf{R}_r)^V$, and $\mathbf{e}_{\omega_r} = \boldsymbol{\omega}_d - \mathbf{R}_d \mathbf{R}_r^T \boldsymbol{\omega}_r$.

In next result we characterize the convergence of the above errors to zero provided that the reference trajectory $\mathbf{q}_r(t)$ is *feasible* and satisfies the additional property that \mathbf{u}_1^r is *sufficiently inside* \mathcal{U}_1 , meaning that there exists a time instant \bar{t} and a positive number ε such that the distance of \mathbf{u}_1^r from the boundary of \mathcal{U}_1 is greater than $\varepsilon > 0$ for all $t > \bar{t}$, i.e.,

$$\text{dist}(\mathbf{u}_1^r(t), \partial \mathcal{U}_1) > \varepsilon, \quad \forall t > \bar{t}. \quad (12)$$

Corollary 1. Assume $\mathbf{q}_r(t)$ is a feasible trajectory and that it satisfies the additional property in (12). Assume that $\mathbf{R}_d(t) \in \mathcal{R}(\mathbf{f}_r(t))$ for any t and that $\boldsymbol{\omega}_d(t)$ and $\dot{\boldsymbol{\omega}}_d(t)$ are well defined for any t . Consider the control \mathbf{u}_1 and \mathbf{u}_2 defined at (4) and (5). Assume that the initial condition satisfies (6) and (7). Then the zero equilibrium of the tracking errors \mathbf{e}_R , \mathbf{e}_ω , \mathbf{e}_p and \mathbf{e}_v is exponentially stable and there exists a time instant $\bar{t} \geq t_0$ such that $\mathbf{e}_{R_r}(t) = \mathbf{e}_{\omega_r}(t) = \mathbf{0}$ for all $t > \bar{t}$. The region of attraction is characterized by (6) and (7).

Remark 1. The proposed controller (in particular the attitude controller (5)) relies on the availability of $\boldsymbol{\omega}_d$, and $\dot{\boldsymbol{\omega}}_d$. These quantities depend in turn on \mathbf{R}_d which is the output of an optimization algorithm executed at each control step. In order for $\boldsymbol{\omega}_d$ and $\dot{\boldsymbol{\omega}}_d$ to be well defined and available the optimization must ensure a sufficient smoothness of \mathbf{R}_d . This could be enforced by adding, e.g., a regularization term in the cost function J . If in the real case at hand this is not possible (or not implementable), then at each time instant in

which \mathbf{R}_d is not smooth the attitude controller will undergo a new transient phase. In practice, see Sec. V, we have experimentally ascertain that the presence of a few isolated non-smooth instants does not constitute at all a real problem for the stability of the implementation and that regularization is actually not needed for practical stabilization.

IV. COMPUTATION OF \mathbf{R}_d FOR AN IMPORTANT CASE

The proposed control method is kept on purpose general regarding two main features: the choice of \mathcal{U}_{xy} in (1) and the choice of the cost function J . The former allows the method to be used for a large set of aerial vehicles with different actuation capabilities. The latter allows the engineer to customize the definition of similarity between two orientations in order to comply with the particular task at hand. In this section we illustrate how these two general features are particularized for a specific meaningful case.

In particular, we consider the case of \mathcal{U}_{xy} defined in (3) and the following choice of cost function J

$$J(\mathbf{R}_r, \mathbf{R}') = 1 - \mathbf{b}_{3r}^T \mathbf{b}'_3, \quad (13)$$

where $\mathbf{R}_r = [\mathbf{b}_{1r} \mathbf{b}_{2r} \mathbf{b}_{3r}]$ and $\mathbf{R}' = [\mathbf{b}'_1 \mathbf{b}'_2 \mathbf{b}'_3]$. The cost function J in (13) is minimized whenever $\mathbf{b}_{3r} = \mathbf{b}'_3$ and maximized whenever $\mathbf{b}_{3r} = -\mathbf{b}'_3$. In the following we show how it is possible to efficiently compute an \mathbf{R}_d that belongs to $\mathcal{R}(\mathbf{f}_r, \mathbf{R}_r)$ and is also equal to \mathbf{R}_r if $\mathbf{R}_r \in \mathcal{R}(\mathbf{f}_r, \mathbf{R}_r)$. These are in fact the requirements needed for \mathbf{R}_d in order for Theorems 1 and Corollary 1 to be valid.

Let us start by instantiating $\mathcal{R}(\mathbf{f}_r)$ for this particular case. From simple geometrical considerations on the cylindrical shape of the set \mathcal{U}_1 it is easy to see that

$$\mathcal{R}(\mathbf{f}_r) = \left\{ \mathbf{R}' \in SO(3) \mid \mathbf{f}_r^T \mathbf{b}'_3 \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2} \right\}, \quad (14)$$

which states that the vector \mathbf{f}_r must lie within the cylinder of radius r_{xy} generated about the axis \mathbf{b}'_3 .

Using (13) and (14) we can rewrite the minimization problem in terms of the only variable \mathbf{b}'_3 , instead of the whole matrix \mathbf{R}' , as

$$\min_{\substack{\mathbf{f}_r^T \mathbf{b}'_3 \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2} \\ \|\mathbf{b}'_3\|^2 = 1}} -\mathbf{b}_{3r}^T \mathbf{b}'_3, \quad (15)$$

where r_{xy}^2 , \mathbf{f}_r^T and \mathbf{b}_{3r}^T are the givens of the problem.

In the case that $\mathbf{f}_r^T \mathbf{b}_{3r} \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2}$ then $\mathbf{b}'_3 = \mathbf{b}_{3r}$ is the solution to (15). Otherwise, let us write \mathbf{b}'_3 as the sum of two components $\mathbf{b}'_3 = \mathbf{b}'_{3\parallel} + \mathbf{b}'_{3\perp}$, where $\mathbf{b}'_{3\parallel}$ is parallel to the plane spanned by \mathbf{b}_{3r} and \mathbf{f}_r , while $\mathbf{b}'_{3\perp}$ is perpendicular to it, i.e., is parallel to $\mathbf{b}_{3r} \times \mathbf{f}_r$. It is easy to see that the cost function in (15), i.e., $\mathbf{b}_{3r}^T \mathbf{b}'_3$, is not affected by $\mathbf{b}'_{3\perp}$, in fact $\mathbf{b}_{3r}^T \mathbf{b}'_3 = \mathbf{b}_{3r}^T \mathbf{b}'_{3\perp} + \mathbf{b}_{3r}^T \mathbf{b}'_{3\parallel} = 0 + \mathbf{b}_{3r}^T \mathbf{b}'_{3\parallel}$. The vector $\mathbf{b}'_{3\parallel}$ can be written using the Rodrigues' rotation formula as $\mathbf{b}'_{3\parallel}(\theta) = \mathbf{b}_{3r} \cos \theta + (\mathbf{k} \times \mathbf{b}_{3r}) \sin \theta + \mathbf{k}(\mathbf{k} \cdot \mathbf{b}_{3r})(1 - \cos \theta)$, where $\mathbf{k} = \frac{\mathbf{b}_{3r} \times \mathbf{f}_r}{\|\mathbf{b}_{3r} \times \mathbf{f}_r\|}$ and θ is the rotation angle that univocally defines $\mathbf{b}'_{3\parallel}$. Noting that the constraint $\|\mathbf{b}'_3\|^2 = 1$ is automatically verified by $\mathbf{b}'_{3\parallel}(\theta)$

Algorithm 1: Bisection used to solve problem (16)

Data: n (number of iterations \propto solution accuracy)
Data: \mathbf{b}_{3r} , \mathbf{f}_r , and r_{xy} (givens of the problem)
1 $\mathbf{k} \leftarrow \frac{\mathbf{b}_{3r} \times \mathbf{f}_r}{\|\mathbf{b}_{3r} \times \mathbf{f}_r\|}$, $\theta_{max} \leftarrow \arcsin(\|\mathbf{k}\|)$, $\theta \leftarrow \theta_{max}/2$
2 **for** $i = 1$ **to** n **do**
3 **if** $\mathbf{f}_r^T \mathbf{b}'_{3\parallel}(\theta) \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2}$ **then** $\theta \leftarrow \theta - \frac{1}{2} \frac{\theta_{max}}{2^i}$
4 **else** $\theta \leftarrow \theta + \frac{1}{2} \frac{\theta_{max}}{2^i}$
5 **return** θ

for any θ , we further simplify (15) in terms of the only scalar variable θ as

$$\min_{\mathbf{f}_r^T \mathbf{b}'_{3\parallel}(\theta) \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2}} -\mathbf{b}_{3r}^T \mathbf{b}'_{3\parallel}(\theta). \quad (16)$$

Given that the feasible set is connected, the minimization problem (16) can be efficiently solved numerically using a bisection method (as the one shown in Algorithm 1).

In order to finally compute \mathbf{R}_d from \mathbf{b}_{3d} we suggest to employ the following formula, as done in [15]: $\mathbf{R}_d = \begin{bmatrix} \underbrace{(\mathbf{b}_{3d} \times \mathbf{b}_{1r}) \times \mathbf{b}_{3d}}_{\mathbf{b}_{1d}} & \underbrace{\mathbf{b}_{3d} \times \mathbf{b}_{1r}}_{\mathbf{b}_{2d}} & \mathbf{b}_{3d} \end{bmatrix}$. Finally, we note that if

$\mathbf{R}_r \in \mathcal{R}(\mathbf{f}_r, \mathbf{R}_r)$ then $\mathbf{f}_r^T \mathbf{b}_{3r} \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2}$ which, as we previously said, implies that $\mathbf{b}_{3d} = \mathbf{b}_{3r}$. Then, it results $\mathbf{R}_d = \mathbf{R}_r$, as asked by the second requirement on the computation of \mathbf{R}_d .

The described algorithm takes a negligible time to be run on a standard computer, thus allowing a real time numerical control implementation at frequencies that are way below 1 ms for each control loop, if needed by the application. In case of different sets \mathcal{U}_{xy} and different cost functions J either similar efficient approaches can be used or the method presented here can be used as a conservative approximation.

V. EXPERIMENTS

1) *Experimental Setup:* The LBF platform used to perform the experiments is the Tilt-Hex robot, an in-house developed fully actuated vehicle (see Fig. 4). The Tilt-Hex is an example of a platform that can be controlled by the full-pose geometric controller presented in Sec. III. We carefully chose the Tilt-Hex to perform the experiments as the platform is able to emulate different values of $r_{xy} \in [0, r_{xy\max}]$ (see (3)), which means that we can emulate seamlessly underactuated and fully actuated platforms. Furthermore, the Tilt-Hex can emulate for validation purposes the cylindric force constraint of (3) by choosing r_{xy} small enough and avoiding trajectories that require extreme values of u_3 , see Fig. 3.

All components of the Tilt-Hex are off-the-shelf available or 3D printable. The diameter of the Tilt-Hex (distance between two propeller hubs) is 0.8m. The total mass, including a 2200mAh battery, is $m = 1.8\text{kg}$. The propellers are tilted first about the axis that connects the rotor with the center of the hexarotor shape ($\alpha = \pm 35^\circ$), then about the axis that is perpendicular to the previous rotation axis and \mathbf{e}_3 ($\beta = 25^\circ$). These angles ensure a balanced selection between large lateral forces and inefficient losses as a result of internal forces.

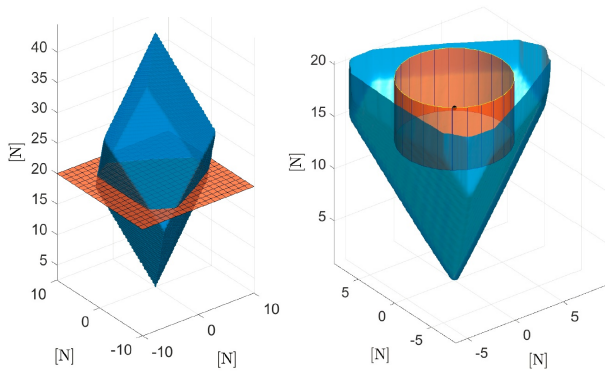


Fig. 3. Left: The blue volume encloses the set of feasible forces at hovering, obeying the constraints of minimal and maximum rotor spinning velocity for the Tilt-Hex. The red plane visualizes the cut of the sectional view of the plot on the right. Right: lower part of the cut of the left figure. The red cylinder visualizes the volume of the imposed cylindric force constraint, whose radius can be shrunk at will to have $r_{xy} \in [0, r_{xy\max}]$. Notice that the cylinder is fully inside the volume of feasible forces. The black dot in the center visualizes the force needed to hover horizontally, i.e., the nominal working point.

The 30.5 cm propeller blades are powered by MK3638 brushless motors provided by MikroKopter. The maximum lift force of a single motor-propeller combination is 12 N. The electronic speed controller (ESC) driving the motor is a BL-Ctrl-2.0 from MikroKopter. The controller, running on the ESC, is an in-house development [21] that controls the propeller rotational speed in closed loop at a variable frequency (e.g., when the propeller rotation speed is 70 Hz the control frequency is 3.29 kHz). An inertial measurement unit (IMU) provides accelerometer and gyroscope measurements (500 Hz) and a marker-based motion capture system provides position and orientation measurements (100 Hz) of the platform. The motion capture and IMU measurements are fused via a UKF state estimator to obtain the full state at 500 Hz.

In the real implementation we have omitted, on purpose, all the terms in which ω_d and $\dot{\omega}_d$ appear in the control (or equivalently, we have considered as if, virtually, $\omega_d = 0$). We have done so in order to assess the level of practical importance of those terms and how the controller is in practice robust to the non-correct evaluation of these terms. As it will be seen next it turned out that the tracking performance is still very good even without those terms, therefore one can still obtain acceptable result by neglecting them. However, as requested from the theory and also seen in preliminary simulations which we performed but are not reported in the paper for lack of space and also because much less significant than real experiments, a perfect tracking is guaranteed only if those terms are considered.

Finally, the controller has been implemented in Matlab-Simulink and runs at 500 Hz on a stationary base PC connected to the Tilt-Hex through a serial cable. As the computational effort of the controller is very low (considerably below 1 ms per control loop) it could be ported easily to an on-board system. Based on our experience with a similar porting, we expect the performances of an onboard implementation to be much better than the Matlab-Simulink implementation, thanks to the possibility of reaching a faster control frequency (larger



Fig. 4. The Tilt-Hex performing Exp. 1.1 at different time instances: Left $t = 15.4$ s; middle $t = 37.7$ s; right $t = 47.9$ s. Although the reference orientation is constant and horizontal the Tilt-Hex adapts its orientation to allow for following the reference position.

than 1 kHz) and almost real-time capabilities (latency below 1 ms). The presence of the cable is clearly disturbing for the tracking purpose since it produces an unmodeled wrench. Furthermore, the oscillations of the cable are rather erratic and therefore hard to model and compensate. Therefore, all considered, the experiments proposed in Section V-2 represent a worst case scenario from this point of view, in the sense that an onboard implementation can only perform better than what we tested.

2) *Experimental Validations:* We conducted three experimental validation campaigns, in which the tasks has been always to follow a given reference $\mathbf{q}_r(t) = (\mathbf{p}_r(t) = [p_{rx}(t) p_{ry}(t) p_{rz}(t)]^T, \mathbf{R}_r(t))$. In the *Experimental Batch 1* the value of r_{xy} in (3) and of \mathbf{R}_r is kept constant and three experiments are performed: Exp. 1.1, Exp. 1.2 and Exp. 1.3, which are detailed in the following. In the Experiment 2, \mathbf{R}_r varies over time. Finally, in the Experiment 3 is r_{xy} that varies over time.

For the reader's visual convenience the rotation matrices used internally by the controller, have been converted in the plots to *roll-pitch-yaw* angles, with the convention $\mathbf{R}_\bullet \rightarrow \phi_\bullet, \theta_\bullet, \psi_\bullet$. In plots where data are very noisy a filtered version (darker color) is presented together with the original data (lighter color in background). We shall refer to the plots in a figure directly by the name of plotted signal, which is easily understandable from the legends present in each plot.

The interested reader is referred to the multimedia attachment of this paper to fully enjoy the videos all the experiments. Furthermore, all the experimental data (with suitable scripts to plot them) are provided for download at the following link <http://homepages.laas.fr/afranchi/files/2017/dataset1.zip>.

3) *Experimental Batch 1:* In this batch we impose $p_{ry}(t) = 0$ m and $p_{rz}(t) = 1$ m, while $p_{rx}(t)$ oscillates sinusoidally between -1.2 m and 1.2 m with time-varying frequency, chosen such that the envelope of $\ddot{p}_{rx}(t)$ is first quasi-linearly increasing from 0 m/s² up to 5.9 m/s² and then quasi-linearly decreasing down to 0 m/s² – see the corresponding signals in Fig. 5. On the other side we set $\mathbf{R}_r(t) = \mathbf{I}_{3 \times 3}$.

In Exp. 1.1 a value of $r_{xy} = 3$ N has been selected, which fits well inside the actual maximum lateral force of the Tilt-Hex given its mass of $m = 1.8$ kg. This means that in the parts of the trajectory in which $|\ddot{p}_{rx}| > \frac{3\text{N}}{1.8\text{kg}} = 1.66$ m/s² we expect the controller to let the platform deviate from \mathbf{R}_r in order to track the high lateral acceleration. On the other side we expect a good independent tracking of position and orientation when $\ddot{p}_{rx} \leq 1.66$ m/s². In fact, Exp. 1.1 is meant to illustrate the canonical behavior of proposed controller when controlling a fully-actuated LBF platform.

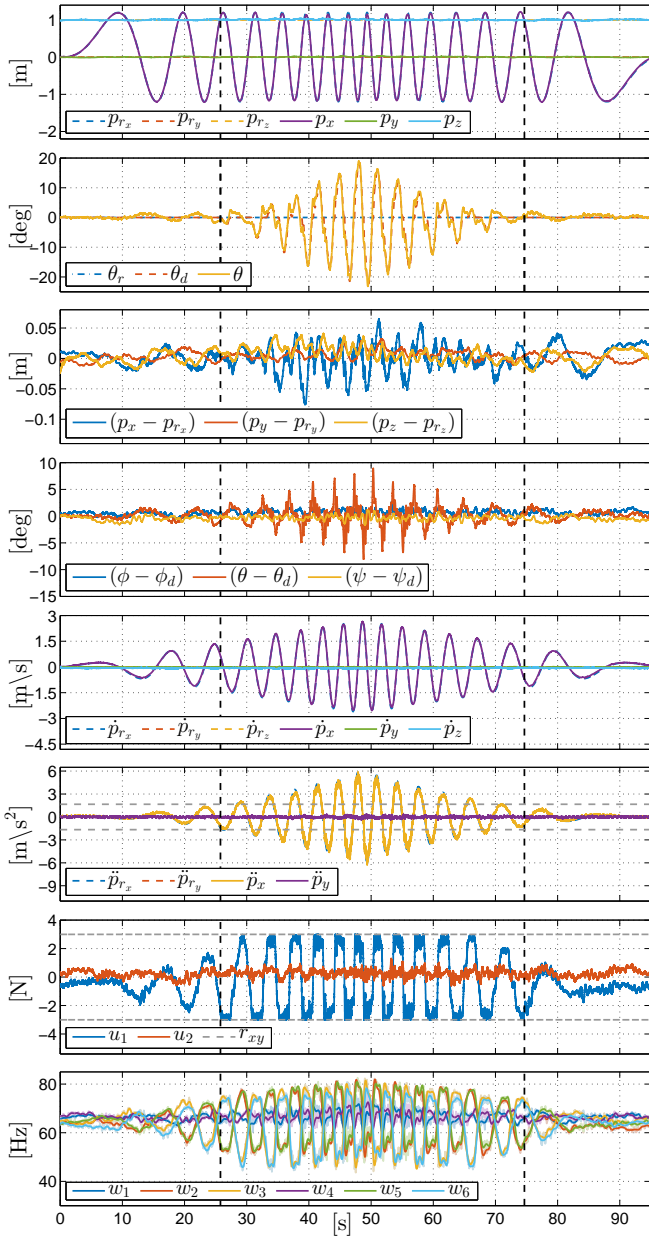


Fig. 5. Exp. 1.1: Desired position: sinusoidal motion along the \mathbf{x}_W axis with constant amplitude and triangular (first increasing then decreasing) frequency. Desired orientation: constantly horizontal. Lateral force bound: constant $r_{xy} = 3\text{N}$.

In Exp. 1.2 we tell the controller that $r_{xy} = 0\text{N}$ (underactuated aerial vehicle). Therefore we expect the controller to let the platform deviate almost always from \mathbf{R}_r . This experiment is meant to show that the proposed controller can handle the (classic) underactuated case, thus not requiring the user to switch between different controllers.

In Exp. 1.3 we replace our controller with the state-of-the-art controller presented in [6], a controller that does not take into account the input saturations. This experiment is meant to show how the proposed controller outperforms the state of the art controllers in terms of robustness and stability.

Experiment 1.1: Fig. 5 visualizes the main results of Exp. 1.1. There are three clearly distinct temporal phases

separated by the vertical dashed lines in the plots and defined by $t \in T_1 = [0\text{s}, 25.8\text{s}]$, $t \in T_2 = [25.8\text{s}, 74\text{s}]$, and $t \in T_3 = [74\text{s}, 95\text{s}]$, respectively.

In the first and third phases $\mathbf{q}_r(t)$ is always feasible. In fact $|\ddot{p}_{rx}|$ is always below 1.66m/s^2 , see discussion above. In the second phase, instead, $\mathbf{q}_r(t)$ is not always feasible and in the middle of the second phase, in the neighborhood of $t = 49\text{s}$, $\mathbf{q}_r(t)$ is mostly unfeasible, since $|\ddot{p}_{rx}|$ has peaks of 5.9m/s^2 (3.5 times the maximum lateral acceleration attainable while keeping a horizontal orientation).

Accordingly to what expected, in the ‘feasible’ phases (T_1 and T_3) both the orientation and position tracking errors w.r.t. $\mathbf{q}_r(t)$, are relatively low. In particular we have $\|\mathbf{p}(t) - \mathbf{p}_r(t)\| < 0.02\text{m}$ and $|\theta(t) - \theta_r(t)| < 1.7^\circ$ and zero average for the two errors. In the ‘unfeasible’ phase (T_2), the position tracking is still good ($\|\mathbf{p}(t) - \mathbf{p}_r(t)\| < 0.06\text{m}$) while $\mathbf{R}_d(t)$ sensibly deviates from $\mathbf{R}_r(t)$ with a peak overshooting 20° for $|\theta_d(t) - \theta_r(t)|$. In fact, tilting is the only way by which the platform can track the desired position, given the lateral force bounds. It is interesting to note that not only the reference position is well tracked along the whole experiment, but also the translational velocity and acceleration are.

One can also appreciate how the controller keeps always the lateral force u_1, u_2 within the requested bounds and at the same time touches and stays on the bounds several times for several seconds. This is a clear index that the controller exploits at best the platform capabilities. Comparing the plots one can notice also how when the lateral force is saturated, the controller exploits the platform tilting in order to compensate for the partial loss of control authority and attain the force required to produce the needed acceleration.

Figure 4 shows the Tilt-Hex performing Exp. 1.1 in three different time instants. Furthermore, we encourage the reader to watch the multimedia attachment showing this and the other experiments. Finally, for the sake of completeness, we present also the actual six rotor spinning frequencies w_1, \dots, w_6 .

Experiment 1.2: To test the behavior of the controller with an underactuated aerial system, in Exp. 1.2 we set $r_{xy} = 0\text{N}$ and let the controller track the same trajectory of Exp. 1.1. In this way the Tilt-Hex should behave like a collinear, underactuated, multi-rotor. Figure 6 shows the main plots, while the plots that are similar to the ones in Fig. 5 are omitted.

Contrarily to Exp. 1.1, in Exp. 1.2, phases T_1 and T_2 do not exist. The whole experiment is a long unfeasible phase due to the constraint $r_{xy} = 0\text{N}$, which makes impossible, at any time, to track the constant $\mathbf{R}_r(t) = \mathbf{I}_{3 \times 3}$ while following the sinusoidal reference position trajectory. The orientation tracking of Exp. 1.2 is compared to the one Exp. 1.1. In the period of time T_1 defined for Exp. 1.1, $|\theta(t) - \theta_r(t)|$ in Exp. 1.2, reaches 11° , i.e., 6.5 times the peak of Exp. 1.1 in the same period. In the period of time T_2 , $|\theta(t) - \theta_r(t)|$ in Exp. 1.2, reaches 31° i.e., about 1.5 times the peak of Exp. 1.1 in the same period. Regarding the translational behavior, the peak of the position tracking error is about 3 times larger (in the period T_1) and 1.4 times larger (in the period T_2), when compared to the error peak of Exp. 1.1 in the same periods. This is due to the fact that full actuation helps in minimizing the position tracking error too. Furthermore, we can see that

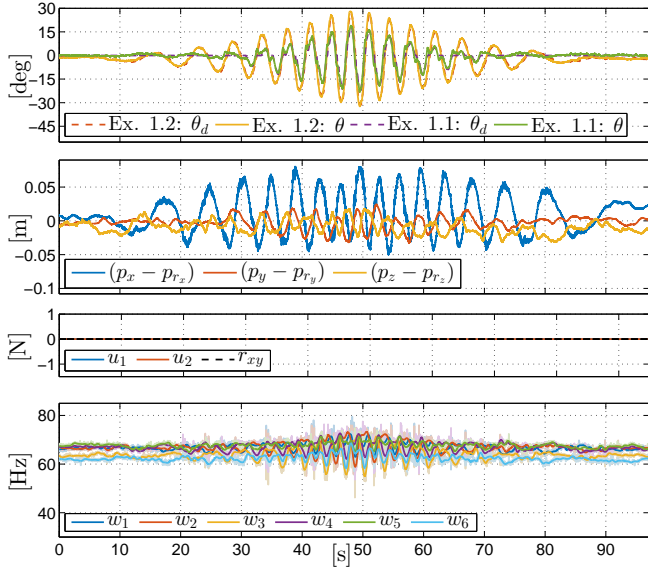


Fig. 6. Exp. 1.2: Same desired trajectory as in Exp.1.1 but with $r_{xy} = 0$.

that the inputs u_1 and u_2 remain zero as expected during the full trajectory tracking, as required. Finally, for completeness, we present also in this case the six rotor spinning frequencies w_1, \dots, w_6 .

Experiment 1.3: In order to compare with the state-of-the-art methods such as [6], in Exp. 1.3 we tested the controller with a saturated rotor spinning velocity with the minimum and maximum values in Exp. 1.1 ($43 \text{ Hz} \leq w_i \leq 83 \text{ Hz}$, $i \in [1 \dots 6]$). The results are depicted in Fig. 7. The platform tracks well the reference trajectory till the input reaches its limit ($t=34 \text{ s}$) (see the dashed horizontal lines in the plot of w_1, \dots, w_6). After that time the controller asks for spinning velocities outside the limits, which are hardly saturated. The trajectory tracking performance decreases rapidly, until the system becomes completely unstable diverging from the reference position ($\|\mathbf{p} - \mathbf{p}_r\| > 0.5 \text{ m}$) and reference velocity ($\|\dot{\mathbf{p}} - \dot{\mathbf{p}}_r\| > 1.2 \text{ m/s}$) such that we had to abort the experiment. This experiment clearly shows how our controller outperforms a state-of-the-art controller in terms of performances and, most important of all, stability and safety.

4) *Experiment 2:* To present the full capabilities of the full pose controller on LBF vehicles, in Exp. 2 we set $\mathbf{p}_r(t)$ as in Exp. 1.1, but we additionally ask the platform to follow a $\mathbf{R}_r(t)$ generated applying to $\mathbf{I}_{3 \times 3}$ a sinusoidal rotation about the \mathbf{y}_W axis (with an amplitude of 10°). This rotational motion is particularly chosen such that the orientation of the Tilt-Hex is in opposition of phase with respect to the orientation that an underactuated vehicle would need in order to track $\mathbf{p}_r(t)$ (i.e., the top part of the platform facing outwards at the two ends of the position trajectory, while for, e.g., a quadrotor the top would face always toward the center of the position trajectory). Also in this case, see Fig. 8, the reference-to-actual position error and the desired-to-actual orientation error remain bounded and small. The maximum lateral thrust is reached sooner than Exp. 1.1 (at $t = 10 \text{ s}$), due to the special inclination required. This results in an earlier adaptation of \mathbf{R}_d . As expected, at

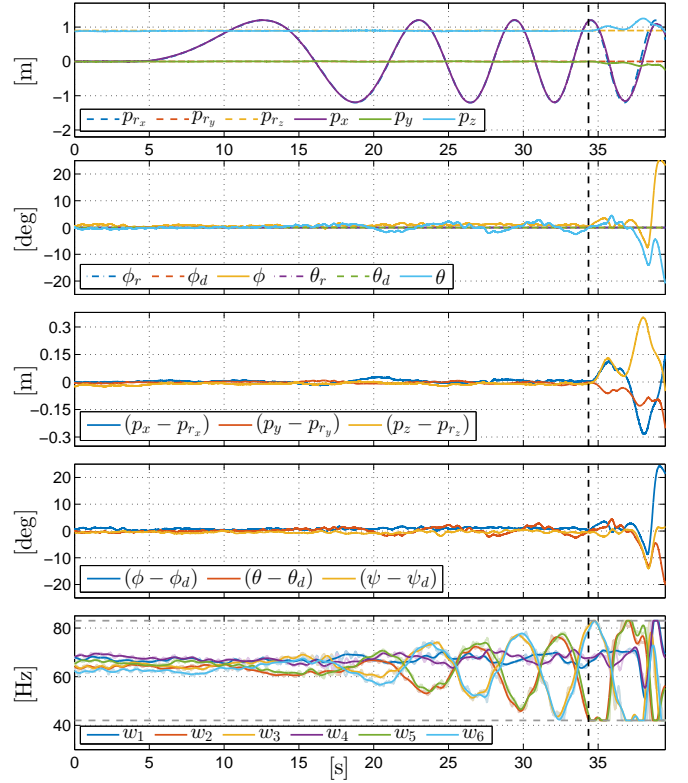


Fig. 7. Exp. 1.3: Same desired trajectory as in Exp.1.1 but with saturated rotor spinning velocity \mathbf{w} (saturation indicated by dashed grey lines in plot five). The experiment is automatically stopped after about 39 s because the system becomes visibly unstable.

the time of highest accelerations ($45 \text{ s} \leq t \leq 55 \text{ s}$) θ_d is almost inverted with respect to θ_r .

5) *Experiment 3:* The conclusive Exp. 3 has been designed to stress the fact that the presented controller can seamlessly work with *under* and *fully-actuated* platforms and moreover with platforms that can actively change between these two configurations during flight, as the one presented in [9]. The plots of are reported in Fig. 9.

The steady state $\mathbf{p}_r(t)$ consists out of two regular sinusoidal motions along the \mathbf{x}_W and \mathbf{y}_W axes with an amplitude of 1.3 m and 0.5 m, respectively, and constant frequencies, while $\mathbf{R}_r(t) = \mathbf{I}_{3 \times 3}$ is constant. The lateral force bound r_{xy} is changed over time, in particular, it is $r_{xy}(t) = 0 \text{ N}$ for $t \in [0 \text{ s}, 18 \text{ s}]$, $r_{xy}(t) = 10 \text{ N}$ for $t \in [38 \text{ s}, 56 \text{ s}]$, and it is linearly increasing from 0 N to 10 N for $t \in [18 \text{ s}, 38 \text{ s}]$.

As it should be, the position tracking is always good. However, initially, the system cannot track at the same time the reference position and the reference orientation. Until $t = 34 \text{ s}$ the system is partially fully actuated (w.r.t. the trajectory to be followed). The orientation tracking gradually improves. At $t = 34 \text{ s}$, r_{xy} is large enough to track the reference orientation at any time. The behavior of r_{xy} is visualized with the dashed lines in the plot of u_1, u_2 , which are always kept within the bounds. The fact that the lateral force bound changes over time does not deteriorate the behavior of the controller, which is instead able to cope with the time-varying constraint exploiting the platform capability always at its best.

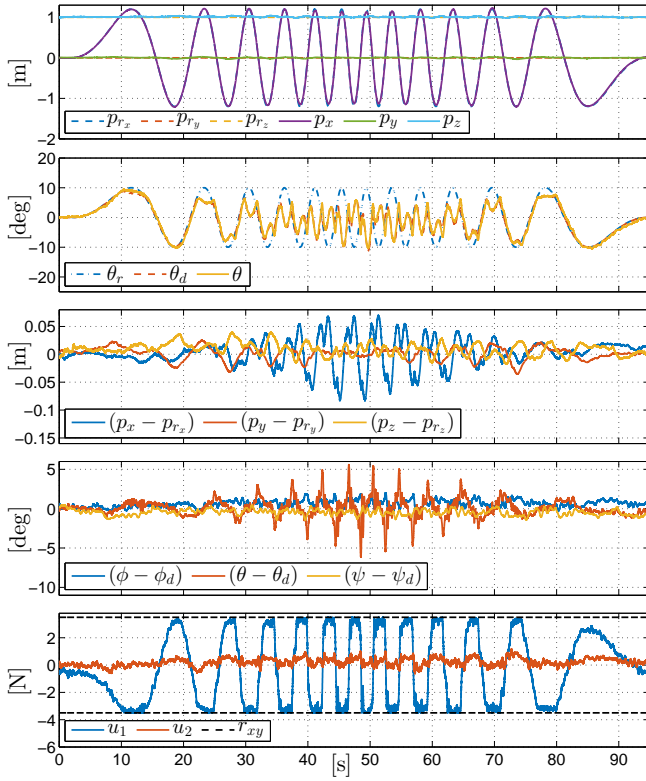


Fig. 8. Exp. 2: Desired position: as in Exp. 1.1. Desired orientation: sinusoidal rotation about the y_w axis in opposition of phase w.r.t. a hypothetical quadrotor following the desired position. Lateral force bound: constant $r_{xy} = 3$ N.

Finally, notice how the ranges of the propeller spinning frequencies utilized by the controller ($w_1 \dots w_6$) naturally increases with the increase of $r_{xy}(t)$ at the benefit of completely tracking the full pose reference trajectory.

VI. CONCLUSIONS

In this paper we introduced the new class of Laterally-Bounded Force (LBF) aerial vehicles. This class is general enough to encompass a large variety of recently conceived aerial vehicles having the possibility to actuate the thrust in a direction other than the principal one. Common underactuated platforms are included in this class as a degenerate (but fully admissible) case. For this class of vehicles we proposed a geometric controller in $SE(3)$ that is able to let it track any feasible full-pose (6D) trajectory. The controller adapts seamlessly to the case the trajectory is (or becomes) not feasible or that the platform is (or becomes) underactuated. Being defined in $SE(3)$ the controller is not prone to the singularities of local chart orientation representations. The practicability of the theory has been shown in real experiments. Furthermore this controller has been already used as inner loop controller for other projects involving also aerial physical interaction, as, e.g., in [10].

In future we plan to study adaptive and robust techniques to deal with parameter uncertainties and malfunction of some of the actuators.

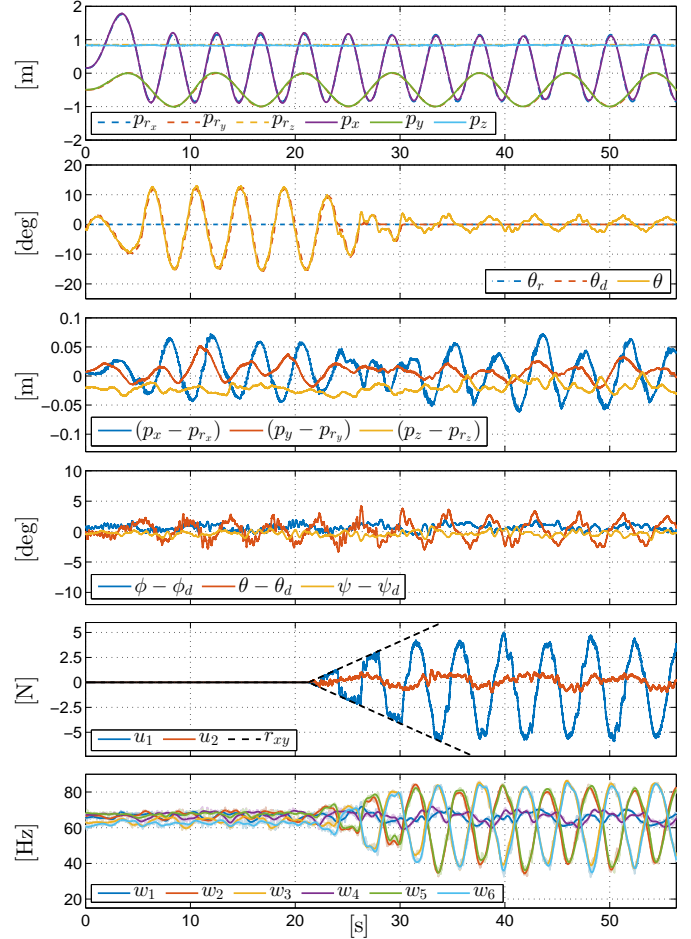


Fig. 9. Exp. 3: Desired position: composition of two sinusoidal motions along x_B and y_B , with constant amplitudes and frequencies. Desired orientation: constant and horizontal. Lateral force bound: r_{xy} linearly increasing from 0 N to 10 N.

REFERENCES

- [1] M. Faessler, A. Franchi, and D. Scaramuzza, "Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 620–626, 2018.
- [2] M. Tognon and A. Franchi, "Dynamics, control, and estimation for aerial robots tethered by cables or bars," *IEEE Trans. on Robotics*, vol. 33, no. 4, pp. 834–845, 2017.
- [3] M. Tognon, S. S. Dash, and A. Franchi, "Observer-based control of position and tension for an aerial robot tethered to a moving platform," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 732–737, 2016.
- [4] M. Tognon, A. Testa, E. Rossi, and A. Franchi, "Takeoff and landing on slopes via inclined hovering with a tethered aerial robot," in *2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Daejeon, South Korea, Oct. 2016, pp. 1702–1707.
- [5] R. Voyles and G. Jiang, "Hexrotor UAV platform enabling dextrous interaction with structures – preliminary work," in *2012 IEEE Int. Symp. on Safety, Security and Rescue Robotics*, College Station, TX, Nov. 2012, pp. 1–7.
- [6] S. Rajappa, M. Ryll, H. H. Bühlhoff, and A. Franchi, "Modeling, control and design optimization for a fully-actuated hexarotor aerial vehicle with tilted propellers," in *2015 IEEE Int. Conf. on Robotics and Automation*, Seattle, WA, May 2015, pp. 4006–4013.
- [7] D. Brescianini and R. D'Andrea, "Design, modeling and control of an omni-directional aerial vehicle," in *2016 IEEE Int. Conf. on Robotics and Automation*, Stockholm, Sweden, May 2016, pp. 3261–3266.
- [8] S. Park, J. J. Her, J. Kim, and D. Lee, "Design, modeling and control of omni-directional aerial robot," in *2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Daejeon, South Korea, 2016, pp. 1570–1575.

- [9] M. Ryll, D. Bicego, and A. Franchi, "Modeling and control of FAST-Hex: a fully-actuated by synchronized-tilting hexarotor," in *2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Daejeon, South Korea, Oct. 2016, pp. 1689–1694.
- [10] M. Ryll, G. Muscio, F. Pierri, E. Cataldi, G. Antonelli, F. Caccavale, and A. Franchi, "6D physical interaction with a fully actuated aerial robot," in *2017 IEEE Int. Conf. on Robotics and Automation*, Singapore, May 2017, pp. 5190–5195.
- [11] V. Mistler, A. Benallegue, and N. K. M'Sirdi, "Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback," in *10th IEEE Int. Symp. on Robots and Human Interactive Communications*, Bordeaux, Paris, France, Sep. 2001, pp. 586–593.
- [12] T. Hamel, R. Mahony, R. Lozano, and J. Ostrowski, "Dynamic modelling and configuration stabilization for an X4-Flyer," in *15th IFAC World Congress*, Barcelona, Spain, 2002, pp. 217–222.
- [13] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *2005 IEEE Int. Conf. on Robotics and Automation*, May 2005, pp. 2247–2252.
- [14] R. Mahony, S.-H. Cha, and T. Hamel, "A coupled estimation and control analysis for attitude stabilisation of mini aerial vehicles," in *2006 Australasian Conf. on Robotics & Automation*, Auckland, New Zealand, Nov. 2006, pp. 3003–3012.
- [15] T. Lee, M. Leoky, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV on $SE(3)$," in *49th IEEE Conf. on Decision and Control*, Atlanta, GA, Dec. 2010, pp. 5420–5425.
- [16] M.-D. Hua, T. Hamel, P. Morin, and C. Samson, "Introduction to feedback control of underactuated VTOL vehicles: A review of basic control design ideas and principles," *IEEE Control Systems Magazine*, vol. 33, no. 1, pp. 61–75, 2013.
- [17] H. Romero, S. Salazar, A. Sanchez, and R. Lozano, "A new UAV configuration having eight rotors: dynamical model and real-time control," in *46th IEEE Conf. on Decision and Control*, New Orleans, LA, Dec. 2007, pp. 6418–6423.
- [18] G. Ducard and M.-D. Hua, "Discussion and practical aspects on control allocation for a multi-rotor helicopter," in *1st Int. Conf. on UAVs in Geomatics*, Zurich, Switzerland, Sep. 2011, pp. 95–100.
- [19] B. Convens, K. Merckaert, M. M. Nicotra, R. Naldi, and E. Garone, "Control of fully actuated unmanned aerial vehicles with actuator saturation," in *20th IFAC World Congress*, Toulouse, France, Jul. 2017, pp. 12 715–12 720.
- [20] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2001.
- [21] A. Franchi and A. Mallet, "Adaptive closed-loop speed control of BLDC motors with applications to multi-rotor aerial vehicles," in *2017 IEEE Int. Conf. on Robotics and Automation*, Singapore, May 2017.