D. Invernizzi, M. Giurato, P. Gattazzo, M. Lovera
Comparison of Control Methods for Trajectory Tracking in Fully Actuated Unmanned Aerial Vehicles
IEEE Transactions on Control Systems Technology, Vol. 29, N. 3, 2021, p. 1147-1160
doi:10.1109/TCST.2020.2992389

Comparison of control methods for trajectory tracking in fully actuated Unmanned Aerial Vehicles

Davide Invernizzi, Mattia Giurato, Paolo Gattazzo, Marco Lovera

***Abstract*—In this paper recently proposed control methods, suitable for trajectory tracking in fully actuated Unmanned Aerial Vehicles (UAVs), are reviewed and experimentally compared. We specifically focus on multirotor platforms with tiltable propellers, for which the thrust and torque delivered by each rotor can be oriented within the aircraft frame by means of servo-actuators. Firstly, the main assumptions underlying the control design model usually employed in the literature are discussed, pointing out the main limitations. Then, some recent control laws that have been proposed to address the trajectory tracking problem for these platforms are reviewed, placing emphasis on their tracking capabilities as well as on the ease of tuning and implementation. Finally, experimental results obtained by applying three control laws on a tilt-arm quadrotor UAV are shown to evaluate and compare their performance. Two experiments, representative of operating conditions for thrust-vectoring UAVs, have been performed: a set-point tracking with level attitude and a full-pose trajectory tracking task.**

***Index Terms*—UAVs, trajectory tracking, fully actuated, multirotor,**

I. Introduction

In recent years the development of multirotor Unmanned Aerial Vehicles (UAVs) with thrust vectoring capabilities has received a growing interest. These systems can achieve a larger degree of actuation compared to coplanar multirotor UAVs since both thrust and torque can be oriented within the airframe. This feature makes thrust-vectoring UAVs capable of performing complex full-pose maneuvers, which is particularly attractive for inspection-like applications that may require, for instance, navigation in a constrained environment. Moreover, being able to deliver both force and torque in any direction enhances the UAV interaction capabilities with the environment, which is especially desirable in aerial manipulation tasks.

Two main technological solutions have been proposed to endow multirotor UAVs with thrust vectoring capabilities: by employing tiltable propellers [1]–[3] and by mounting the propellers in a fixed, non-coplanar fashion [4]–[6]. As per the first kind of UAVs, the tilting of propeller groups is usually achieved by means of servo-actuators. Although being mechanically complex, this configuration is more efficient, in terms of power consumption, than the fixed-tilted one, as it does not lead to the generation of internal forces in hovering with zero-level attitude. Furthermore, the typical fixed-tilted multirotor UAV [5] is not fully actuated since its design is based on a trade-off between efficiency and maneuverability which results in a limited thrust vectoring capability. In this paper we focus on tiltable propellers UAVs which can be considered as fully actuated platforms, as documented in recent works [7], [8].

The main contribution of this work is the evaluation of control architectures which have flown on-board tiltable propellers UAVs on both theoretical and practical aspects. For tiltable propellers platforms, one can consider rotor angular velocities and servo-actuators angles as physical inputs, under the assumption that there are sufficiently fast low level controllers capable of tracking any desired set-point. Except few works [9]–[12], the majority of control designs consider the UAV as a rigid body endowed with an actuation mechanism capable of producing any wrench, *i.e.*, the platform is assumed to be fully actuated. Based on this, the standard approach splits the control design in two parts. First, a control law in charge of computing the wrench required for full-pose tracking is developed according to different strategies (*e.g.*, feedback linearization, cascade control, Lyapunov-based design, *etc.*). Then, an allocation algorithm is used to assign the physical inputs required to deliver the control wrench, based on a nonlinear mapping that depends upon some basic information about the UAV. Although some care must be taken to deal with kinematic singularities and with the limited actuators bandwidth, the input mapping guarantees full actuation in most operating conditions, provided that the tiltable propellers are suitably placed. This strategy decouples the design of the control law and the choice of a specific actuation mechanism. Indeed, the control laws considered in this work can be applied to any fully actuated multirotor UAV under some assumptions that are discussed in detail. A different approach is pursued by [1], in which the proposed design combines dynamic extension and feedback linearization. This strategy results in a more complex control architecture than the one discussed above as it requires integration of intermediate control variables, namely, servo-actuators angular velocities and rotors angular accelerations, to obtain the physical inputs to be sent to the actuators controllers. Moreover, the dynamic extension approach needs linear as well as angular acceleration measurements to compute the control inputs, which are not directly available. To overcome such issues, we propose an alternative feedback linearization approach, inspired by [5], that does not make use of dynamic extension. The quaternion-based P/PID-like control architecture implemented in the PX4 autopilot [13] for underactuated platforms has been adapted to fully actuated ones and has been considered for comparison together with two other nonlinear tracking controllers. For the last two, the strategies presented in our preliminary work [3] and in [8] are considered. While the latter one combines a standard stabilizer for position tracking [14] and a nonlinear

D. Invernizzi, M. Giurato, P. Gattazzo, M. Lovera are with Department of Aerospace Science and Technology, Politecnico di Milano, Via La Masa 34, 20156, Milano, Italy. Email to: {davide.invernizzi, mattia.giurato, paolo.gattazzo, marco.lovera}@polimi.it

cascade controller for attitude tracking [15], the solution proposed in [3] is based on a geometric PID controller for attitude tracking [16] and a quasi-time optimal stabilizer [17] augmented with a conditional integrator [18] for position tracking. The use of nonlinear control laws is motivated by the non-trivial operating conditions in which fully actuated UAVs can be employed, involving possibly large initial errors and challenging tracking tasks that combine attitude and position motion. In such scenarios it is mandatory to guarantee global tracking properties while accounting for saturation of the actuators, in particular, for the limited control force that can be delivered by propellers.

Among the discussed control strategies, the quaternion-based P/PID-like control architecture [13], the modified feedback linearization controller [1], [5] and the nonlinear controller of [3] have been tested experimentally on a tilt-arm quadrotor UAV, a multirotor platform that has four tiltable propellers, developed with low cost off-the-shelf components. The results collected in two experiments are presented to compare the control laws and assess their performance: an aggressive stabilization task, in which the UAV is required to go to a specific position with level attitude, and a full pose trajectory tracking task, in which the UAV has to follow a straight line back and forth with a predefined time law, while keeping a non-null pitch angle.

Notation. In this paper $\mathbb{R}(\mathbb{R}_{>0}, \mathbb{R}_{\geq 0})$ denotes the set of real numbers (positive, nonnegative real numbers), $\mathbb{R}^n$ denotes the n-dimensional Euclidean space and $\mathbb{R}^{m\times n}$ the set of $m \times n$ real matrices. Given $x \in \mathbb{R}^n$, $\|x\| := \sqrt{x_1^2 + \ldots + x_n^2}$ is the Euclidean norm while $\|x\|_\infty := \sup_{t\geq 0} \|x(t)\|$ is the infinity norm. Given $A \in \mathbb{R}^{n\times n}$, we use the compact notation $A \in \mathbb{R}^{n\times n}_{\mathbb{R}}$ to represent a positive definite matrix. The i-th vector in canonical basis of $\mathbb{R}^n$ is denoted as e_i (1 in the i-th position and zeros elsewhere) for $i = 1, \ldots, n$ and the identity matrix in $\mathbb{R}^{n\times n}$ is denoted as $I_n := [e_1 \cdots e_i \cdots e_n]$. The set $\mathrm{SO}(3) := \{R \in \mathbb{R}^{3\times 3} : R^\top R = I_3, \det(R) = 1\}$ denotes the third-order Special Orthogonal group while $\mathbb{S}^1 := \mathbb{R} \bmod 2\pi$ denotes the unit circle manifold. The notation $R_u(\theta)$ is used to represent the rotation matrix corresponding to a rotation around a unit vector $u \in \mathbb{R}^3$ of an angle $\theta \in \mathbb{S}^1$. The normalized attitude distance on $\mathrm{SO}(3)$ with respect to the identity element is denoted as $\|R\|_{\mathrm{SO}(3)} := \sqrt{\frac{1}{4}\mathrm{tr}(I_3 - R)} \in [0, 1]$ for $R \in \mathrm{SO}(3)$. The map $S(\cdot) : \mathbb{R}^3 \to \mathfrak{so}(3)$, given by

$$\omega \mapsto \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{1}$$

defines an isomorphism between $\mathbb{R}^3$ and the vector space of third-order skew-symmetric matrices, *i.e.*, $\mathfrak{so}(3) := \{W \in \mathbb{R}^{3\times 3} : W = -W^\top\}$, such that, one has $S(\omega)y = \omega \times y$, $\forall \omega, y \in \mathbb{R}^3$, where $\times$ is the cross product. The corresponding inverse is $S^{-1}(\cdot) : \mathfrak{so}(3) \to \mathbb{R}^3$. We use the compact notation $\mathrm{c}(\cdot)$, $\mathrm{s}(\cdot)$ for $\cos(\cdot)$ and $\sin(\cdot)$ respectively. The set of piece-wise continuous and bounded function is denoted as $\mathcal{L}_\infty$ while the set of continuously differentiable function up to the *nth* order is denoted as $\mathcal{C}^n$.

II. The tiltable propellers multirotor UAV

The tiltable propellers UAV is a system made by a central rigid body and n rigid links each of which carries a propeller group. The propeller groups can be tilted independently by means of servo-actuators attached to the main body (tilt-arm) or to the propeller group support itself (tilt-rotor)[1]. The dynamics of these systems evolves on the product manifold $\mathrm{SO}(3) \times \mathbb{R}^3 \times (\mathbb{S}^1 \times \mathbb{S}^1)^n$, for which, according to [20], one refers to $\mathrm{SO}(3) \times \mathbb{R}^3$ as the base space whereas to $(\mathbb{S} \times \mathbb{S}^1)^n$ as the shape space. The base space is the manifold in which the main body motion takes place and can be described by the configuration of a body-fixed frame $\mathcal{F}_B = (O_B, \{b_1, b_2, b_3\})$ with respect to an inertial frame $\mathcal{F}_I = (O_I, \{i_1, i_2, i_3\})$. To this end, the rotation matrix $R := [b_1\ b_2\ b_3] \in \mathrm{SO}(3)$, where $b_i \in \mathbb{S}^2$ is i-th unit vector of the body frame resolved in $\mathcal{F}_I$, is used to describe orientation of $\mathcal{F}_B$ with respect to $\mathcal{F}_I$, and the vector $x := [x_1\ x_2\ x_3]^\top \in \mathbb{R}^3$, where $x_i \in \mathbb{R}$ is the i-th inertial component of x, is used to describe the position of O_B with respect to O_I. The relative configuration of the i-th tilt-arm with respect to the main body belongs to the shape space and is parametrized by the angles $(\theta_{a_i}, \theta_{r_i}) \in \mathbb{S}^1 \times \mathbb{S}^1$, where θ_{a_i} describes the inclination of the i-th arm axis with respect to a fixed direction in the body frame and θ_{r_i} is similarly defined for the i-th rotor (see Figure 1). The (body) angular and (inertial) translational velocity are denoted as $\omega \in \mathbb{R}^3$ and $v \in \mathbb{R}^3$, while the relative angular velocity of the i-th arm and rotor are, respectively, $\dot\theta_{a_i} := \omega_{a_i} \in \mathbb{R}$ and $\dot\theta_{r_i} := \omega_{r_i} \in \mathbb{R}_{\geq 0}$[2]. For a tiltable propeller UAV, the tilt angles $\theta_a := (\theta_{a_1}, \ldots, \theta_{a_n}) \in \mathbb{S}^1 \times \cdots \times \mathbb{S}^1$ and the rotor angular velocities $\omega_r := (\omega_{r_1}, \ldots, \omega_{r_n}) \in \mathbb{R}_{\geq 0} \times \cdots \times \mathbb{R}_{\geq 0}$ are controlled by the acutators. Assuming linear dynamics for the servo-actuators and the motors closed-loop systems, the actuators dynamics is described by

$$\theta_{a_i} = G_s(s)\theta^d_{a_i} \tag{2}$$

$$\omega_{r_i} = G_m(s)\omega^d_{r_i} \tag{3}$$

where $\theta^d_{a_i}$ and $\omega^d_{r_i}$ are the desired tilt angle and rotor velocity, which are the control inputs of the system. By compactly collecting the shape space terms $\theta_s := (\theta_{a_1}, \ldots, \theta_{a_n}, \theta_{r_1}, \ldots, \theta_{r_n})$ and $\omega_s := (\omega_{a_1}, \ldots, \omega_{a_n}, \omega_{r_1}, \ldots, \omega_{r_n})$, the dynamical system of the (rigid) tiltable propellers UAV is described by the following set of equations:

$$\dot x = v \tag{4}$$

$$\dot R = RS(\omega) \tag{5}$$

$$\dot\theta_r = \omega_r \tag{6}$$

$$m\dot v - mRS(x_c)\dot\omega = -mge_3 - mRS(\omega)^2 x_c + R(f_c + f_e) \tag{7}$$

$$J(\theta_s)\dot\omega + mS(x_c)R^\top \dot v = -S(\omega)J(\theta_s)\omega - mgS(x_c)R^\top e_3 + \tau_c + \tau_e. \tag{8}$$

[1]The considered configuration is different from the one of *variable pitch* quadrotors [19] in which servo-actuators are used to vary the pitch of the rotor blades and not the orientation of the whole propeller groups. Using pitch variations rather than motor rpm variations makes it possible to generate thrusts with a much wider bandwidth but it does not give full actuation capabilities to UAVs.

[2]We limit our attention to the common case of unidirectional propellers, *i.e.*, propellers endowed with motors that cannot revert the spin direction.

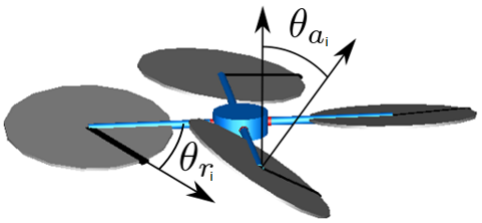

Figure 1. Tiltable propellers quadrotor UAV - tilt and rotor angles definition.

where $g = 9.81\text{ms}^{-1}$ is the gravitational acceleration, $x_c \in \mathbb{R}^3$ is the center of mass location with respect to O_B (resolved in $\mathcal{F}_B$), $m \in \mathbb{R}_{>0}$ is the overall mass of the vehicle, and $J(\theta_s) = J(\theta_s)^\top \in \mathbb{R}^{3\times3}_{>0}$ is the shape-dependent inertia matrix resolved in $\mathcal{F}_B$ with respect to O_B. Furthermore $f_e := f_e(R, x, \omega, v, \dot{\omega}, \dot{v}, \theta_s, \omega_s, \dot{\omega}_s, t)$ and $\tau_e := \tau_e(R, x, \omega, v, \dot{\omega}, \dot{v}, \theta_s, \omega_s, \dot{\omega}_s, t)$ are the *body* force and torque, respectively, which account for the inertial terms, related to the tilt-arms, not included in $S(\omega)J(\theta_s)\omega$, as well as for aerodynamic forces and torques acting on the UAV main body[3]. Finally, $f_c := f_c(\theta_a, \omega_r) \in \mathbb{R}^3$ and $\tau_c := \tau_c(\theta_a, \omega_r) \in \mathbb{R}^3$ represent the force and torque delivered at O_B by the rotors, *i.e.*:

$$f_c(\theta_a, \omega_r) := \sum_{i=1}^{n} R_{bp_i}(\theta_{a_i}) f_{p_i}(\omega_{r_i}) \tag{9}$$

$$\tau_c(\theta_a, \omega_r) := \sum_{i=1}^{n} S(x_{br_i}(\theta_{a_i})) R_{bp_i}(\theta_{a_i}) f_{p_i}(\omega_{r_i}) + \tau_{p_i}(\omega_{r_i}) \tag{10}$$

where $f_{p_i}(\omega_{r_i})$ and $\tau_{p_i}(\omega_{r_i}) \in \mathbb{R}^3$ are, respectively, the force and torque delivered by the *i-th* rotor resolved in the propeller frame $\mathcal{F}_{p_i}$, $R_{pi}(\theta_{a_i}) \in \text{SO}(3)$ is the rotation matrix describing the orientation of the *i-th* propeller frame with respect to $\mathcal{F}_B$ and $x_{br_i}(\theta_{a_i}) \in \mathbb{R}^3$ is the position vector from O_B to the hub of the *i-th* rotor disk, resolved in $\mathcal{F}_B$. With respect to the fixed-tilted configuration in which x_{br_i} is constant in $\mathcal{F}_B$, the control wrench (9)-(10) becomes a nonlinear function of the variables (θ_a, ω_r) for the tiltable propellers platform.

III. Simplified model for control design

In this section we review the simplified dynamical model that is exploited for control design purposes in most of the works that deal with thrust-vectoring UAVs ([1], [5], [8]).

A. Main assumptions

We start by presenting the main assumptions under which the full set of equations (4)-(8) can be approximated to have a sufficiently accurate representation of the system dynamics, in the flight regime of interest, and a manageable control problem.

Assumption 1. *Besides the assumption that the UAV components are rigid, we also consider the following:*

1) *the UAV is operated at low speed ($\|v\|_\infty \leq 3-4\text{m/s}$);*
2) *shape-changes due to the actuation mechanism (servo-actuators) do not affect significantly the geometric and dynamical properties of the system;*
3) *the point O_B coincides with the UAV center of mass;*
4) *the control wrench $w_c := (f_c, \tau_c)$ delivered by the actuators spans $\mathbb{R}^3 \times \mathbb{R}^3$,* i.e.*, the system is fully actuated;*
5) *the actuator dynamics is sufficiently faster than the expected system dynamics;*
6) *full state measurements are available.*

The first assumption is reasonable for an indoor flight scenario as the one considered in this work. It is also the case for most outdoor operating conditions involving multirotor UAVs, *e.g.*, for surveillance, inspections, photography, etc.. The second assumption is admissible for the typical configuration (see, *e.g.*, Figure 5) in which the center of mass of each propeller groups lies almost on the axis of rotation of the corresponding arm and when the angular momenta of the rotors are sufficiently small. Then, assuming that O_B coincides with the UAV center of mass (assumption three), equations (7)-(8) take the following form:

$$m\dot{v} = -mge_3 + Rf_c \tag{11}$$

$$J\dot{\omega} = -S(\omega)J\omega + \tau_c \tag{12}$$

where we denoted with $J := J(\theta_s = 0)$ the nominal inertia of the UAV. The fourth assumption is valid when the propeller groups are at least three ($n \geq 3$) and properly spaced, as will be shown in Section III-B. In such case, one can directly use the wrench w_c as the input variable for control design, so that the problem reduces to the one of controlling a rigid body affected by constant disturbances. This approach has the advantage of being independent from the specific actuation mechanism: the input allocation problem, *i.e.*, the computation of the physical inputs (θ_a^d, ω_r^d) from w_c, can be solved independently. Clearly, for this strategy to work, the closed-loop response of the servo-actuators and of the motors should be fast enough to track the desired input commands so that the control wrench can be almost instantaneously delivered, *i.e.*, $G_m(s) = G_s(s) = 1$, in a sufficiently large bandwidth (assumption five). Note that in this case one can consider $\theta_a^d = \theta_a$ and $\omega_r^d = \omega_r$. The last assumption is required since most of the control laws exploit a full state feedback strategy to address the trajectory tracking problem. Full state information can be obtained by employing suitable state estimation techniques that exploit the available pose and velocity measurements, typically provided by a motion capture system (laboratory setup) or by on-board sensors (outdoor conditions).

By referring to the simplified dynamical model described by equations (11)-(12), one sees that the attitude dynamics (12) evolves independently from the translational one (11). Moreover, since the control force f_c can be delivered in any direction with respect to the airframe (thanks to full actuation) and R is measured, it is possible to use the virtual input $f_d := Rf_c$ as control variable for the position dynamics. By direct substitution, equation (11) can be written as follows:

$$m\dot{v} = -mge_3 + f_d. \tag{13}$$

[3]For the sake of conciseness we do not report here the (complicated) expressions of f_e, τ_e and $J(\theta)$ as they are not instrumental for our discussion.

Therefore, the control design for attitude and position tracking can be performed separately. An alternative but coupled model could be obtained by expressing (13) in terms of the body velocity $v_b := R^\top \dot{x}$: this approach could have the advantage of enabling the derivation of control laws that have invariant properties with respect to the choice of the inertial frame, at the expense of an increased complexity [21].

Remark 1. *Assumption 1 allows writing the dynamics of the tiltable propeller UAV as the one of a fully actuated rigid body. Although the wrench (f_e, τ_e) can be neglected under Assumption 1, the map from the physical inputs to the delivered wrench,* i.e., *$(\theta_a, \omega_r) \mapsto w_c(\theta_a, \omega_r)$, can induce a significant disturbance wrench due to modeling uncertainties when used in an allocation algorithm (more details about this in Remark 4).*

B. Simplified model of the wrench map

In this section the input allocation problem is addressed, pointing out the related challenges. Simultaneously, we review one of the most common allocation algorithm (see [3], [8]) that applies to the typical configuration with n tiltable propellers placed on a common plane and angularly equispaced.

As we mentioned in Section II, it is difficult to obtain the exact model of the map from the rotor states to the propeller wrench (f_{p_i}, τ_{p_i}). Thus one usually consider an approximate model of this map that works in a sufficiently broad range of operating conditions. To this end, most works on small-scale multirotor UAV assume that f_{p_i} and τ_{p_i} are orthogonal to the corresponding rotor disk plane and employ a quadratic model to describe their relationship with the spinning rate ω_{r_i}, *i.e.*,

$$f_{p_i} := k_f \omega_{r_i}^2 e_3 \tag{14}$$

$$\tau_{p_i} = -\varepsilon_i k_\tau \omega_{r_i}^2 e_3, \tag{15}$$

where $k_f, k_\tau \in \mathbb{R}_{>0}$ are the thrust and torque coefficient, respectively (which can be obtained experimentally in static conditions) and $\varepsilon_i \in \{1, -1\}$ defines the rotation direction. Although this model is valid when considering small deviations from the hovering condition [22], it has been successfully exploited in dynamic and even in highly acrobatic maneuvering, as documented by several experimental works [23], [24]. Then, under the assumptions that the rotor hubs are placed at a distance $\ell \in \mathbb{R}_{>0}$ from the body center of mass in the $\{b_1, b_2\}$ plane, each making an angle $\gamma_i \in \mathbb{S}^1$ with respect to b_1, the wrench map $(\theta_a, \omega_r) \mapsto w_c(\theta_a, \omega_r)$ can be written as:

$$w_c(\theta_a, \omega_r) = \begin{bmatrix} \sum_{i=1}^n k_f \omega_{r_i}^2 R_{bp_i}(\theta_a) e_3 \\ \sum_{i=1}^n \left(k_f \omega_{r_i}^2 S(x_{br_i}) R_{bp_i}(\theta_a) e_3 - \varepsilon_i k_\tau \omega_{r_i}^2 R_{bp_i}(\theta_a) e_3 \right) \end{bmatrix} \tag{16}$$

where $x_{br_i} := \ell R_{e_3}(\gamma_i) e_1$, $R_{bp_i} := R_{e_3}(\gamma_i) R_{e_1}(\theta_{a_i})$, $R_{e_1}(\theta_{a_i})$ is the rotation matrix associated to the *i-th* tilt angle. Let now

$$u := (T_1, \dots T_n, \theta_{a_1}, \dots, \theta_{a_n}) \in \mathcal{U}, \tag{17}$$

denote the control input, where the input set has been defined as

$$\mathcal{U} := \underbrace{\mathbb{R} \times \dots \times \mathbb{R}}_{\text{n copies}} \times \underbrace{\mathbb{S}^1 \times \dots \times \mathbb{S}^1}_{\text{n copies}} \tag{18}$$

and the input variable ω_{r_i} has been substituted with the corresponding thrust T_i by virtue of (14). One effective way of writing (16) is to highlight the contribution of the components (in the propeller frame) of the force produced by each rotor, *i.e.*, $R_{e_1}(\theta_{a_i}) k_f \omega_{r_i}^2 e_3 = R_{e_1}(\theta_{a_i}) T_i e_3 = \begin{bmatrix} 0 & T_i \mathrm{c}(\theta_{a_1}) & T_i \mathrm{s}(\theta_{a_i}) \end{bmatrix}^\top$, which appear linearly in (16). In particular, by defining a vector-valued function $f_u : \mathcal{U} \to \mathbb{R}^{2n}$ as

$$f_u(u) := \begin{bmatrix} T_1 \mathrm{c}(\theta_{a_1}) & \cdots & T_n \mathrm{c}(\theta_{a_n}) & T_1 \mathrm{s}(\theta_{a_1}) & \cdots & T_n \mathrm{s}(\theta_{a_n}) \end{bmatrix}^\top, \tag{19}$$

the mapping from f_u to w_c is compactly expressed as:

$$w_c := M(\ell, \sigma, \varepsilon_1, \dots, \varepsilon_n) f_u(u) \tag{20}$$

where $\sigma := \frac{k_\tau}{k_f}$ and

$$M := \begin{bmatrix} \mathrm{s}(\gamma_1) & 0 & \cdots & \mathrm{s}(\gamma_n) & 0 \\ -\mathrm{c}(\gamma_1) & 0 & \cdots & -\mathrm{c}(\gamma_n) & 0 \\ 0 & 1 & \cdots & 0 & 1 \\ -\varepsilon_1 \sigma \mathrm{s}(\gamma_1) & \ell \mathrm{s}(\gamma_1) & \cdots & -\varepsilon_n \sigma \mathrm{s}(\gamma_n) & \ell \mathrm{s}(\gamma_n) \\ \varepsilon_1 \sigma \mathrm{c}(\gamma_1) & -\ell \mathrm{c}(\gamma_1) & \cdots & \varepsilon_n \sigma \mathrm{c}(\gamma_n) & -\ell \mathrm{c}(\gamma_n) \\ -\ell & -\varepsilon_1 \sigma & \cdots & -\ell & -\varepsilon_n \sigma \end{bmatrix}. \tag{21}$$

Assumption 2. *The physical properties of the UAV that parametrize map M defined in equation* (21)*, namely, $n, \ell, \sigma, \varepsilon_1, \dots, \varepsilon_n$, are chosen such that $\mathrm{rank}(M) = 6$.*

Note that Assumption 1 is easily verified by typical configurations, *e.g.*, the tilt-arm tricopter [2], the tilt-arm/rotor quadcopter [1], [3], the tilt-rotor hexacopter [8]. In such case, the linear system (20) admits ∞^{2n-6} solutions of the form:

$$f_u(u) = M^+ w_c + \left(I_{2n} - M^+ M \right) w, \tag{22}$$

in which $M^+ = M^\top \left(M M^\top \right)^{-1}$ is the right pseudo-inverse of M and $w \in \mathbb{R}^{2n}$ is an arbitrary vector, projected onto the null space of M by the projection operator $I_{2n} - M^+ M$. Note that once $f_u(u)$ is recovered from (22), system (19) is made of n independent subsystems of the form:

$$\begin{aligned} f_{u_{i+1}} &= T_i \mathrm{c}(\theta_{a_i}) \\ f_{u_i} &= T_i \mathrm{s}(\theta_{a_i}). \end{aligned} \tag{23}$$

In order to retrieve the inputs ω_{r_i} and θ_{a_i}, the following control allocation algorithm can be employed $\forall T_i \neq 0$:

$$\begin{aligned} T_i &= \pm \sqrt{f_{u_i}^2 + f_{u_{i+1}}^2} \\ \omega_{r_i} &= \sqrt{\tfrac{T_i}{k_f}} \\ \mathrm{c}(\theta_{a_i}) &= \tfrac{f_{u_{i+1}}}{T_i} \\ \mathrm{s}(\theta_{a_i}) &= \tfrac{f_{u_i}}{T_i}, \end{aligned} \tag{24}$$

for $i = 1, 2, \dots, n$. Therefore, when $T_i \neq 0$, the *i-th* system (23) admits two solutions and in (24) the one corresponding to $T_i > 0$ must be selected, so that the inputs are feasible according to (14), *i.e.*, $\omega_{r_i} > 0 \ \forall i$. As the physical command required by the servo-actuator controller is an angle, one can use the four-quadrant inverse tangent $\mathrm{atan2} : \mathbb{R}^2 \setminus \{0,0\} \mapsto [-\pi, \pi)$ to retrieve the tilt-angle θ_{a_i}:

$$\theta_{a_i} := \mathrm{atan2}(f_{u_i}, f_{u_{i+1}}). \tag{25}$$

It is worth mentioning that the solution $f_u^\star = M^+ w_c$ corresponding to $w = 0$ in (22) is the minimum-norm solution of system (20). Since by definition $\|f_u\|^2 = \sum_{i=0}^n T_i^2$, the

minimum-norm solution naturally yields a power efficient result, preventing unnecessary tilting of the arms that would just produce a balanced internal wrench.

Remark 2. ***Kinematic singularities of the wrench map.*** *Whenever $T_i = 0$, which is the solution corresponding to $f_{u_i} = f_{u_{i+1}} = 0$, the* i-th *system* (23) *is indeterminate: any $\theta_{a_i} \in \mathbb{S}^1$ is a possible solution. Note that this is a possibly disruptive condition since for $T_i \approx 0$ numerical errors would produce unpredictable values for θ_{a_i}: the system is ill-conditioned. To gain a better insight into this issue, let us consider the tangent map from the tangent space of the input manifold to the tangent space relative to the system dynamics,* i.e., *$D_u w_c : (\mathbb{R})^{2n} \to \mathbb{R}^6$, which can be computed as follows:*

$$D_u w_c(u) = M D_u f_u(u) \tag{26}$$

where

$$D_u f_u(u) := \begin{bmatrix} c(\theta_{a_1}) & 0 & 0 & -T_1 s(\theta_{a_1}) & 0 & 0 \\ & \ddots & & & \ddots & \\ 0 & & 0 & 0 & & 0 \\ 0 & 0 & c(\theta_{a_n}) & 0 & 0 & -T_n s(\theta_{a_n}) \\ s(\theta_{a_1}) & 0 & 0 & T_1 c(\theta_{a_1}) & 0 & 0 \\ & \ddots & & & \ddots & \\ 0 & & 0 & 0 & & 0 \\ 0 & 0 & s(\theta_{a_n}) & 0 & 0 & T_n c(\theta_{a_n}) \end{bmatrix}. \tag{27}$$

By direct computation, the determinant of the matrix in equation (27) *is the product of the propellers thrusts,* i.e., *$\det(D_u f_u) = T_1 \cdots T_n$. As expected, whenever $T_i = 0$, $\det(D_u f_u) = 0$ and the system becomes ill-conditioned. To handle such undesired conditions, known as "kinematic singularities" in the literature, one possible strategy is to employ $w \in \mathbb{R}^{2n}$, defined in* (19)*, to produce "null motions",* i.e., *different values of $f_{u_i}, f_{u_{i+1}}$ that produce the same control wrench w_c as the minimum norm solution.*

Remark 3. ***Limited actuators bandwidth.*** *The use of a static allocation algorithm is reasonable when the time variation of the required control wrench w_c is sufficiently slow with respect to the actuators dynamics ((point 5) of Assumption 1). Clearly, combining a too aggressive controller and significant initial pose errors would require fast and large tilt-angle variations θ_a, which could be impossible to track: the underlying trade-off between performance and stability should be considered in the control design validation. Clearly, including the actuators dynamics* (2)-(3) *in the control design itself would be more accurate; however, it would also make the overall architecture more complex than needed, at least for standard applications [11]. Note, in passing, that motors and servo-actuators have similar bandwidth for the current application although servos have a more complex dynamical behavior than motors (see Section VI-A and [1]).*

Remark 4. *The wrench map defined by equation* (20) *is based on the exact knowledge of the relative position of the center of mass with respect to the propellers. Since this is not the case in practice, one should consider $x_{br_i}(\theta_{a_i}) = \bar{x}_{br_i}(\theta_{a_i}) - x_c$, where $x_c \in \mathbb{R}^3$ is the error on the knowledge of the center of mass location in the body frame: as a result, since $\sum_{i=1}^n R_{bp_i}(\theta_{a_i}) f_{p_i}(\omega_{r_i}) =: f_c$, there will be a disturbance torque*

$$\Delta\tau_c(\theta_a, \omega_r) := -\sum_{i=1}^{n} S(x_c) R_{bp_i}(\theta_{a_i}) f_{p_i}(\omega_{r_i}) = -S(x_c) f_c \tag{28}$$

acting on the platform. Furthermore, four additional effects should be considered as well: 1) each propeller delivers a different thrust magnitude T_i for the same ω_{r_i} due to unavoidable mechanical differences; 2) the thrusts f_{pi} are not exactly directed along the assumed directions due to mounting errors; 3) the interaction of the spinning rotor with air produces a component of f_{pi} also in the plane of the rotor; 4) the spinning rates ω_{r_i} are assigned by inverting the simplified quadratic model (14)-(15)*, which is exact only in hovering conditions. Therefore, the actual wrench delivered by each propeller can be generally written as $(f_{p_i}, \tau_{p_i}) = (T_i e_3, -\varepsilon_i \sigma T_i) + (\Delta f_{p_i}, \Delta\tau_{p_i})$ so that the overall disturbance wrench associated with the propeller wrench is*

$$\Delta w_c(\theta_a, \omega_r) = \begin{bmatrix} \sum_{i=1}^n R_{bp_i}(\theta_{a_i}) \Delta f_{p_i} \\ -S(x_c) f_c + \sum_{i=1}^n \left(S(x_{br_i}) R_{bp_i}(\theta_{a_i}) \Delta f_{p_i} + R_{bp_i}(\theta_{a_i}) \Delta\tau_{p_i} \right) \end{bmatrix} \tag{29}$$

in which we neglected second order terms involving products of x_c, $\Delta f_{p_i}, \Delta\tau_{p_i}$. While neglecting exogenous aerodynamic effects may be reasonable for the considered flight conditions, the disturbance wrench (29) *can have a significant impact on the UAV stability and must be addressed by the control design, either by removing upfront the issue (using a better estimate of the wrench map and more reliable actuators) or by employing a dynamic controller to reject or at least mitigate the effect of such a disturbance.*

IV. Control problem: full pose tracking

In this section we present the full pose tracking problem for the system described by equations (4)-(8). The control objective is to make the main body of the UAV track a sufficiently smooth trajectory $t \mapsto (R_d(t), x_d(t), \omega_d(t), v_d(t))) \in \mathrm{SO}(3) \times \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^3$, where $\omega_d(t) = (R_d^\top(t) \dot{R}_d(t))^\vee$ is the desired *body* angular velocity and $v_d(t) = \dot{x}_d(t)$ is the desired *inertial* translational velocity. The steady-state wrench, obtained by inverting the system dynamics evaluated on the desired reference, is:

$$f_d^{ss}(t) := m(\dot{v}_d(t) + g e_3) \tag{30}$$

$$\tau_c^{ss}(t) := J\dot{\omega}_d(t) + S(\omega_d(t)) J \omega_d(t). \tag{31}$$

It is usually assumed that the steady state wrench is bounded, *i.e.*, $f_d^{ss}(\cdot), \tau_c^{ss}(\cdot) \in \mathcal{L}_\infty$, in order to have a well-posed tracking problem in the presence of actuators saturations. In this regard, the following common assumption is made.

Assumption 3. *The desired trajectory $t \mapsto (R_d(t), x_d(t), \omega_d(t), v_d(t)) \in \mathrm{SO}(3) \times \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^3$ is such that $v_d(t) = \dot{x}_d(t) \in \mathcal{C}^1$ and that $\omega_d(t) = (R_d^\top(t) \dot{R}_d(t))^\vee \in \mathcal{C}^1 \cap \mathcal{L}_\infty$. Furthermore, the desired angular and translational acceleration are bounded,* i.e., *$\dot{\omega}_d(t), \dot{v}_d(t) \in \mathcal{L}_\infty$.*

V. Overview of existing control strategies

By relying on the full actuation of the platform, most of the control strategies developed for trajectory tracking in thrust-vectoring UAVs (*e.g.*, [5], [8]) split the control design in two blocks, as already mentioned in Section III. In the first block the control wrench w_c is computed to obtain desirable tracking performance and robustness. The second block comprises the wrench mapper, which is in charge of control allocation, *i.e.*, the computation of the commands to be sent to the actuators. In this way, the control law can be designed independently from the specific platform configuration as long as the full-actuation assumption is granted and the actuators are fast enough. Nonetheless, we start by recalling one of the first strategies reported in the literature that has been experimentally validate, which is based on dynamic extension and feedback linearization [1]. To this regard, we propose a similar but alternative control law (still based on feedback linearization) which fits within the standard paradigm, is computationally lighter, and requires less feedback information. Then, we consider a version of the quaternion-based P/PID-like controller for underactuated multirotor platforms [13], adapted to deal with fully actuated ones. Finally, we present two more recent controllers that address the tracking problem from a nonlinear point of view.

A. The approach of Ryll et al. [1] (FDBL)

The paper [1] was one of the first to propose a control strategy for a tilt-rotor quadcopter UAV ($n = 4$). The main peculiarity of the approach presented therein is that rotor angular accelerations $\dot{\omega}_{r_i}$ and servo-actuators tilting velocities ω_{a_i} are used as variables for control design. As they do not appear directly in the equations of motion (11)-(12) combined with (16) one has to resort to a dynamic extension procedure. First of all, the dynamical model (11)-(12) is additionally simplified by neglecting the gyroscopic term $S(\omega)J\omega$ and the disturbance wrench w_e. Then, the input $u_a := \begin{bmatrix} \omega_{a_1} & \cdots & \omega_{a_4} \end{bmatrix}^\top$ is fictitiously included as follows:

$$\begin{bmatrix} \ddot{x} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} ge_3 \\ 0_{3\times 1} \end{bmatrix} + \begin{bmatrix} \frac{1}{m}R & 0_{3\times 3} \\ 0_{3\times 3} & J^{-1} \end{bmatrix} \begin{bmatrix} B_f(\theta_a) & 0_{3\times 4} \\ B_\tau(\theta_a) & 0_{3\times 4} \end{bmatrix} \begin{bmatrix} \omega_r \\ u_a \end{bmatrix} = \begin{bmatrix} ge_3 \\ 0_{3\times 1} \end{bmatrix} + J_R J_{u_a}(\theta_a) \begin{bmatrix} \omega_r \\ u_a \end{bmatrix}, \quad (32)$$

where $B_f(\theta_a)$ and $B_\tau(\theta_a)$ are the input matrices that relate the column matrix $u_r := \begin{bmatrix} \omega_{r_1}^2 & \cdots & \omega_{r_4}^2 \end{bmatrix}^\top$ of squared angular velocities to force and torque, respectively. Matrix $J_R J_{\theta_a}$ in equation (32) has rank four: the input-output map can be inverted only at a higher differential level where the input u_a explicitly appear. Specifically, by differentiating equation (32), one gets:

$$\begin{bmatrix} \dddot{x} \\ \ddot{\omega} \end{bmatrix} = A(R, \theta_a, \omega_r) \begin{bmatrix} u_r \\ u_a \end{bmatrix} + d(\theta_a, \omega_r, R, \omega), \quad (33)$$

in which matrix $A(R, \theta_a, \omega_r)$ has $\text{rank}(A) = 6$ as long as $\omega_{r_i} \neq 0$ for $i = 1, \ldots, n$ (see [1]). Therefore, feedback linearization may be applied to equation (33) by selecting:

$$\begin{bmatrix} u_r \\ u_a \end{bmatrix} = A^+ \left(\begin{bmatrix} v_x \\ v_R \end{bmatrix} - d \right) + (I_8 - A^+ A)z \quad (34)$$

where A^+ is the pseudo-inverse of matrix A, v_x and v_R are virtual inputs and the last term $(I_8 - A^+A)z$ corresponds to the projection of z onto the null space of A. The presence of a $2-$dimensional null space for matrix A is a consequence of the system over-actuation as already seen in Section III-B. By substituting (34) in (33), one gets:

$$\dddot{x} = v_x \quad (35)$$

$$\ddot{\omega} = v_R. \quad (36)$$

Then in [1], the virtual inputs are selected as follows:

$$v_x := \dddot{x}_d - K_{x_1}\dot{e}_v - K_{x_2}e_v - K_{x_3}e_x \quad (37)$$

$$v_R := \ddot{\omega}_d - K_{\omega_1}\dot{e}^\ell_\omega - K_{\omega_2}e^\ell_\omega - K_{\omega_3}e^\ell_R, \quad (38)$$

where $K_{x_i}, K_{\omega_i} \in \mathbb{R}^{3\times 3}_{>0}$, $i = 1, 2, 3$, are diagonal matrices and

$$\text{(pos.)} \begin{cases} e_x := x - x_d \\ e_v := v - v_d \end{cases} \quad \text{(att.)} \begin{cases} R^r_e := R^\top_d R \\ e^\ell_\omega := \omega - \omega_d. \end{cases} \quad (39)$$

are the tracking errors [4]. By substituting (37)-(38) in (35)-(36), the closed-loop dynamics reads:

$$\dddot{e}_x = -K_{x_1}\ddot{e}_x - K_{x_2}\dot{e}_x - K_{x_3}e_x \quad (40)$$

$$\ddot{e}^\ell_\omega = -K_{\omega_1}\dot{e}^\ell_\omega - K_{\omega_2}e^\ell_\omega - K_{\omega_3}e^r_R \quad (41)$$

where $e^r_R := \frac{1}{2}S^{-1}\left(R^r_e - R^{r\top}_e\right)$ is the left trivialized derivative of the navigation function $\Psi(R^r_e) := \frac{1}{2}\text{tr}(I - R^r_e)$ [26].

Remark 5. *In [1] it is claimed that the gain matrices K_{ω_i}, $i = 1, 2, 3$, should be selected to make $\lambda^3 + K_{\omega_{1ii}}\lambda^2 + K_{\omega_{2ii}}\lambda + K_{\omega_{3ii}}$ $(i = 1, 2, 3)$ Hurwitz polynomials. However, the claim is valid only for small angular velocity and attitude errors because for large attitude errors $\dot{e}^r_R = \frac{1}{2}\left(tr(R^{r\top}_e)I_3 - R^{r\top}_e\right)e^r_\omega \neq e^\ell_\omega$, where we defined $e^r_\omega := \omega - R^{r\top}_e\omega_d$. Therefore, one cannot study the global stability properties of the closed-loop system by referring to the dynamical model* (40)-(41)*. However, for small attitude and angular velocity errors,* i.e.*, when $R^r_e \approx I_3$ and $e_\omega \approx 0$, one can consider the approximation $\dot{e}^r_R \approx e^r_\omega \approx e^\ell_\omega$. In such case, the closed-loop dynamics would be described by*

$$\dddot{e}^r_R = -K_{\omega_1}\ddot{e}^r_R - K_{\omega_2}\dot{e}^r_R - K_{\omega_3}e^r_R \quad (42)$$

and (local) exponential convergence can be concluded for a suitable choice of the gain matrices.

Remark 6. *The control law* (34) *requires $2n$ integrators to compute the physical inputs (ω_r, θ_a), which increases the overall computational load with respect to the static input allocation proposed in Section III-B. As shown in [Section VI-A] [1], it is also advisable to employ an optimization strategy, by exploiting vector z in* (34)*, in order to reduce the power loss. Otherwise, after maneuvering, the UAV could reach hovering with inclined propellers: in this condition, a lot of power is wasted in producing internal forces which are not used to balance gravity.*

Based on the above discussion and inspired by [5], we will consider a simplified feedback linearization controller that exploits the control wrench as control input and the

[4] In this work we exploit the nomenclature "right" and "left" for attitude errors as proposed in [25, page 554].

invertibility of the wrench map according to the procedure outlined in Section III-B. Specifically, by defining

$$f_c := mR^\top (ge_3 + v_x) \tag{43}$$
$$\tau_c := S(\omega)J\omega + Jv_R, \tag{44}$$

the linearization of the dynamics is achieved at a lower differential level with respect to (35)-(36):

$$\ddot{x} = v_x \tag{45}$$
$$\dot{\omega} = v_R. \tag{46}$$

Note, however, that the attitude system is still nonlinear when one includes the kinematic equation (5). The virtual inputs are selected as a proportional-integral stabilizer with feedforward:

$$v_x := \ddot{x}_d - K_{x_1} e_v - K_{x_2} e_x - K_{x_3} e_{I_p} \tag{47}$$
$$v_R := S^\top(e^r_\omega) R^{r\top}_e \omega_d + R^{r\top}_e \dot{\omega}_d - K_{\omega_1} e^r_\omega - K_{\omega_2} e^r_R - K_{\omega_3} e_{I\omega} \tag{48}$$
$$\dot{e}_{Ip} = e_x \tag{49}$$
$$\dot{e}_{I\omega} = e^r_R \tag{50}$$

where $K_{x_i}, K_{\omega_i} \in \mathbb{R}^{3\times 3}_{>0}$, $i = 1,2,3$, are suitably selected diagonal gain matrices. The corresponding closed-loop system is:

$$\dot{e}_x = e_v \tag{51}$$
$$\dot{e}_v = -K_{x_1} e_v - K_{x_2} e_x - K_{x_3} e_{Ip} \tag{52}$$
$$\dot{e}_{Ip} = e_x \tag{53}$$
$$\dot{R}^r_e = R^r_e S(e^r_\omega) \tag{54}$$
$$\dot{e}^r_\omega = -K_{\omega_1} e^r_\omega - K_{\omega_2} e^r_R - K_{\omega_3} e_{I\omega} \tag{55}$$
$$\dot{e}_{I\omega} = e^r_R. \tag{56}$$

While the position closed-loop is linear, the attitude one can be linearized for small errors by employing the approximations $R^r_e \approx I_3 + S(\delta\theta)$, $e^r_\omega \approx \delta e_\omega$, where $\delta\theta \in \mathbb{R}^3$ and $\delta e_\omega \in \mathbb{R}^3$. The linearized closed-loop reads:

$$\dot{\delta\theta} = \delta e_\omega \tag{57}$$
$$\dot{\delta e}_\omega = -K_{\omega_1} \delta e_\omega - K_{\omega_2} \delta\theta - K_{\omega_3} e_{I\omega} \tag{58}$$
$$\dot{e}_{I\omega} = \delta\theta \tag{59}$$

where the approximation $e^r_R \approx \frac{1}{2} S^{-1}\left((I_3 + S(\delta\theta) - (I_3 - S(\delta\theta))\right) = \delta\theta$ has been exploited for small attitude errors. Based on the above model, one can apply any suitable tuning method to compute the gain matrices according to desired closed-loop performance.

B. Quaternion-based P/PID controller (CPID)

This control strategy is based on a nested loops architecture with a proportional action in the outer loop and a PID controller in the inner loop for both attitude and position, following the quite established cascaded scheme exploited, *e.g.*, by the PX4 autopilot for multirotor UAVs [13].

The attitude control law is defined by (see also Figure 2):

$$\omega_v := -K^A_{po} \mathrm{sgn}(\mathrm{q}_e) \mathbf{q}_e \tag{60}$$
$$\tau_c := K^A_{ff}\omega_v + \left(K^A_{pi} + K^A_i \tfrac{1}{s}\right)(\omega_v - \omega) - K^A_d \tfrac{s}{1+\frac{s}{N}}\omega, \tag{61}$$

where $\mathbf{q}_e \in \mathbb{R}^3$, $\mathrm{q}_e \in \mathbb{R}$ are, respectively, the vectorial and the scalar part of the quaternion error $q_e \in \mathbb{S}^3$, which is computed as the Hamiltonian product between the desired quaternion $q_d \in \mathbb{S}^3$and the conjugate of the measured quaternion $q \in \mathbb{S}^3$, *i.e.*, $q_e := q_d \otimes q^*$ [15]. Furthermore, K^A_{po} is the proportional gain of the outer loop, K^A_{ff}, K^A_{pi}, K^A_i and K^A_d are the feedforward, the proportional, the integral and the derivative gains of the inner loop respectively.

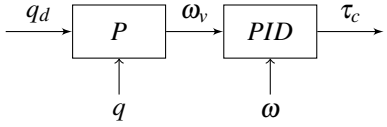

Figure 2. Nested attitude control loops

The position controller, as can be seen from Figure 3, has a nested architecture with the same structure of the attitude controller:

$$v_v := -K^P_{po} e_x, \tag{62}$$
$$f_c := R^\top \left(K^P_{ff} v_v - \left(K^P_{pi} + K^P_i \tfrac{1}{s}\right)(v - v_v) - K^P_d \tfrac{s}{1+\frac{s}{N}} v + mge_3\right). \tag{63}$$

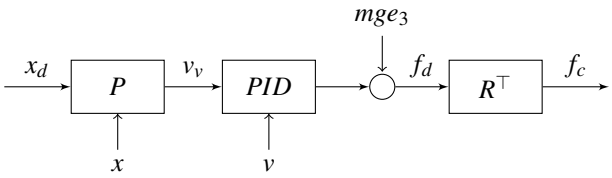

Figure 3. Nested position control loops

Remark 7. *The controller* (60)-(63) *is designed for a stabilization rather than a tracking task and its performance with respect to time-varying references might not be so satisfactory and it depends heavily on the tuning. When assuming near hovering conditions,* $\mathbf{q}_e \approx \frac{1}{2}\delta\theta$*, and the linearized closed-loop is described by three independent equations, one for each axis. Although the number of parameters involved in the control law is large, tuning techniques such as structured* $\mathcal{H}_\infty$ *can be employed to achieve satisfactory performance in terms of set-point tracking and disturbance rejection capabilities. The outer attitude loop* (60) *computes the desired angular velocity, which is sent as a reference to the inner loop* (61)*, based on the quaternion error and a switching strategy to avoid the unwinding phenomenon [15]. The use of the* sgn *function in* (66) *to avoid unwinding may result in chattering due to noise [27], although the discrete implementation should mitigate such misbehavior [15]. Finally, note that this control design relies on the assumption of negligible gyroscopic effect which is reasonable for small scale UAVs.*

C. The approach of Kamel et al. [8]

The paper [8] presents the mechanical design and the control strategy for a hexacopter tilt-rotor UAV. The controller is made

by a proportional integral stabilizer with feedforward action for position tracking and is based on a cascade architecture for attitude tracking, as shown in Figure 4. In particular, the control wrench is assigned as follows:

$$f_c := R^\top(-k_x e_x - k_v e_v - k_I e_I + f_d^{ss}) \tag{64}$$
$$\dot{e}_I := e_x \tag{65}$$
$$\omega_v := -k_d \mathrm{sgn}(\mathrm{q}_e)\mathbf{q}_e \tag{66}$$
$$\tau_c := S(\omega)J\omega + k_\omega(\omega_v - \omega) + S(x_c)f_c \tag{67}$$

where k_x, k_v and k_I are the proportional and integral gains, respectively, while $\mathbf{q}_e \in \mathbb{R}^3$, $\mathrm{q}_e \in \mathbb{R}$ are defined as in Section V-B. The term $S(x_c)f_c = x_c \times f_c$ in (67) accounts for the offset x_c of the center of mass with respect to the point of application of the control force. For what concerns control allocation, the approach considers the same pseudo-inverse based procedure outlined in Section III. However, the strategy is modified to exclude certain servo-actuators and the corresponding propellers in order to avoid an unwanted chattering behavior in near singular configurations.

Remark 8. *The attitude control law* (66)-(67) *accounts for the disturbance wrench associated with the non-perfect knowledge of center of mass location (Remark 4) by exploiting the additional term $S(x_c)f_c$: this is equivalent to having a better estimate of the center of mass location. Note that x_c is unknown and thus represents an additional term to be tuned (with no clear rule). It is also worth remarking that there is no integral action in the attitude controller* (67) *which may be troublesome when propellers are not perfectly balanced (Remark 4). Note, in passing, that in near hovering conditions the disturbance torque $S(x_c)f_c \approx mgS(x_c)R^\top e_3$ is approximately constant for slowly varying attitude maneuvers and an integral action would also be beneficial to reject such a disturbance. Finally, since the attitude outer loop* (66) *has the form of equation* (60)*, considerations reported in Remark 7 apply here as well.*

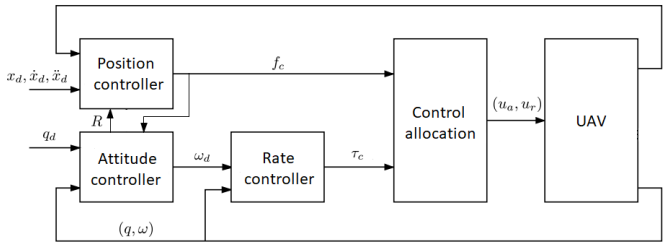

Figure 4. Control scheme proposed in [8].

D. The approach of Invernizzi et al. [3] (IQTO)

In this section we present the control strategy proposed in [3], which exploits the decoupled structure of the equations of motion on $\mathrm{SO}(3) \times \mathbb{R}^3$, as described in Section III. The control law is based on the use of conditional integrators and nonlinear stabilizers, specifically a Quasi Time-Optimal law for position tracking and a geometric PID for attitude tracking. With respect to the other methods described in this work, this architecture guarantees boundedness of the control force as well as global position tracking. Attitude tracking is instead guaranteed almost globally following the proof of [28], which is the best result one can hope for with a continuous time-invariant stabilizer. For a more detailed introduction to these control techniques, the reader is referred to [17], [18]. The control wrench is given by:

$$f_c := R^\top\left(\beta\left(e_x, e_v, e_I\right) + m\left(\dot{v}_d + g e_3\right)\right), \tag{68}$$
$$\ddot{e}_{I_i} := -k_{I_{d_i}}\dot{e}_{I_i} + \mathrm{sat}_{M_{I_1}}\left(k_{I_{p_i}}\left(-e_{I_i} + \mathrm{sat}_{M_{I_2}}(\bar{e}_{x_i})\right)\right) \tag{69}$$
$$\tau_c := -R_d^\top e_R^\ell - K_\omega e_\omega^\ell - K_I e_I + J\dot{\omega}_d + S(\omega_d)J\omega \tag{70}$$
$$\dot{e}_I := R_d^\top e_R^\ell + K_{\omega_I} e_\omega^\ell \tag{71}$$

for $i = 1, 2, 3$, where $\beta_i(e_{x_i}, e_{v_i}, e_{I_i}) := \mathrm{sat}_{\bar{M}_i}\left(u_i(\bar{e}_{x_i}, \bar{e}_{v_i}) - \ddot{e}_{I_i}\right.$, $u_i(\bar{e}_{x_i}, \bar{e}_{v_i}) := -k_{x_i}\left(\bar{e}_{x_i} + \bar{e}_{v_i}\max\left(\frac{|\bar{e}_{v_i}|}{2\bar{M}_i}, \frac{k_{v_i}}{k_{x_i}}\right)\right)$ and $\mathrm{sat}_M(\cdot)$ is the standard symmetric saturation function, *i.e.*, given $s \in \mathbb{R}$, $\mathrm{sat}_M(s) := \max(-M, \min(M, s))$. Herein, $\bar{e}_{x_i} := e_{x_i} + e_{I_i}$, $\bar{e}_{v_i} := e_{v_i} + \dot{e}_{I_i}$ are auxiliary tracking errors, $k_{x_i}, k_{v_i}, k_{I_{d_i}}, k_{I_{p_i}}$ are scalar gains and $\bar{M}_i := M_i - 2M_{I_{1i}}$ are auxiliary saturation bounds, where $M_{I_{1i}}, M_{I_{2i}}$ are suitably selected scalars [3], [18].

Remark 9. *The main challenge in applying the control law* (68)-(71) *is the tuning phase, due to the large number of parameters involved and the non-straightforward interpretation of their contribution on the system response. To this end, one can try to linearize the plant and the control laws for small tracking errors and then to apply suitable tuning techniques (*e.g.*, structured $\mathcal{H}_\infty$). Note that due to the specific choice of the attitude feedback, the linearized control law is time-varying and one has to consider constant reference trajectories to have a linear time-invariant closed-loop system.*

VI. Experimental comparison

In this section we present the results obtained by applying three different control schemes to a tilt-arm quadrotor prototype that was developed by Aerospace Systems and Control Laboratory (ASCL) at Politecnico di Milano (Figure 5) [29]. Among the control designs that have been presented, the quaternion-based P/PID control architecture (60)-(63), the modified feedback linearization controller (43)-(44), (47)-(48) and the nonlinear controller of (68)-(71) have been tested. The design presented in Section V-C was not included in the comparison since we could not find a suitable value for x_c to be used in the disturbance compensation term $S(x_c)f_c$ and without which the UAV could not be flown (see Remark 4).

Two experiments are presented: an aggressive stabilization task characterized by large control inputs, in which the UAV is required to go to a specific position with level attitude starting from a misplaced position, and a full pose trajectory tracking task, in which the UAV has to follow a straight line back and forth with a predefined time law, while keeping a non-null pitch angle.

A. Prototype and indoor setup description

The experimental campaign took place indoor, with attitude and position measured by a motion capture system whereas translational and angular velocity obtained by fusing the

Figure 5. Tilt-arm quadrotor prototype used in the experiments.

measurements of on-board sensors and the position provided by the motion capture system. The prototype used in the experiments is shown in Figure 5. The Flight Control Unit, the servo-motors, the battery and the Electronic Speed Controllers (ESCs) are mounted on the central body. Four tilt-arms are connected to the central body through the servo-motors, thanks to which they can be tilted about their longitudinal axis. The propeller groups (motor and rotor) are fixed at the end of each arm.

The inertial properties of the prototype have been estimated by means of identification experiments carried out on a test-bed. In particular, the pitch inertia moment was identified and the same value was assumed for the roll inertia moment thanks to the almost symmetric configuration. No experiment was performed to evaluate the yaw inertia moment. The inertial and geometric properties of the prototype are collected in Table I, together with the thrust and torque coefficients (k_f, k_τ) that are used in the approximated model of the wrench map (21).

The motors are DC Brushless type, model HP2814 from RCTimer High Performance Series. Thanks to the identification campaigns performed in [30], the static relation between throttle $(\%)$ and angular velocity ω_r^d is expressed as

$$\omega_{r_i}^d = mTh\% + q \tag{72}$$

where $m = 6.031$ and $q = 80.49$. The dynamic model of the motors is expressed by a first order transfer function

$$\bar{G}_m(s) := \frac{\omega_{r_i}(s)}{Th\%} = \frac{a_0}{s + a_1} \tag{73}$$

where $a_0 = 94.55$ and $a_1 = 18.18$ and combining (72)-(73) one obtains the transfer function $G_m(s)$ from the commanded angular velocity to the one actually delivered as defined in (3). The servo-motors (model HS-485HB) are endowed with an internal control loop to track an angle set-point in the range $[-90, 90]$ deg. For this reason the system cannot be considered fully actuated but it can be reasonably assumed as such in a broad range of operating conditions. In order to characterize the dynamics of the servos, in [29] an identification campaign has been performed from which the following transfer function was obtained:

$$G_s(s) = \frac{\theta_{a_i}(s)}{\theta_{a_i}^d(s)} = \frac{b_0}{s^3 + b_1 s^2 + b_2 s + b_3} \tag{74}$$

where θ_{a_i} and $\theta_{a_i}^d$ are the actual and desired tilt angle, respectively and $b_0 = 4670.2519$, $b_1 = 28.355992$, $b_2 = 598.45913$ and $b_3 = 4650.2325$. The actuators bandwidth is around $20\text{rad}/\text{s}$ for both motors and servomotors (Figure 6).

Remark 10. *The considered platform is not endowed with servo-actuators capable of a full rotation. Nonetheless, controllers for fully actuated platforms can be tested by considering admissible full-pose maneuvers,* i.e.*, maneuvers which nominally do not require the actuators to go beyond their saturation limits. For instance, while it is not possible to perform a* 90 deg *rotation on the spot, our platform can still track non-trivial position trajectories while performing attitude maneuvers up to about* 30 deg. *In our case this limitation depends on the maximum thrust that the propellers of our prototype can produce with respect to its weight. In general, to understand if a tilt-arm platform can hover at a desire inclination, one can exploit the allocation algorithm* (22)-(24) *with the commanded wrench* $w_c = \begin{bmatrix} 0 & 0 & mgR_d^T e_3 & 0 & 0 & 0 \end{bmatrix}^\top$, *where* R_d *is the rotation matrix associated with desired pitch/roll angles, to check that the required propeller thrusts and tilt-angles are feasible. Clearly, this procedure depends on the specific allocation algorithm for overactuated UAVs in which there are infinite solutions to* (20) *(see the comment below equation* (25)*).*

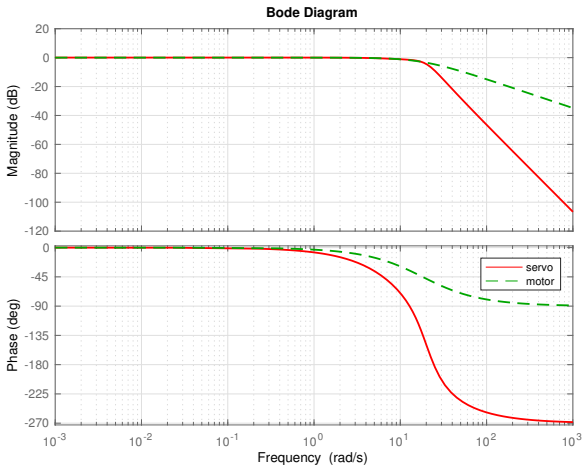

Figure 6. Bode diagrams of the actuators.

B. Controller tuning

The parameter tuning for the IQTO and the FDBL has been performed by requiring similar performance to a step reference (in terms of settling time and overshoot) applied to a simulation platform that was developed by relying on the identified parameters of the prototype. Since the yaw moment of inertia was not identified, the corresponding gains have been adjusted by repeated experiments until a satisfactory behavior

Description	Symbol	Quantity	Unit
Total mass of tilt-rotor	m	2.1	kg
Arm length	ℓ	0.275	m
Inertia around roll axis	J_{xx}	0.0074	$\mathrm{kg}\cdot\mathrm{m}^2$
Inertia around pitch axis	J_{yy}	0.0074	$\mathrm{kg}\cdot\mathrm{m}^2$
Inertia around yaw axis	J_{zz}	–	$\mathrm{kg}\cdot\mathrm{m}^2$
Thrust coefficient	k_f	$2.4619\cdot10^{-5}$	$\mathrm{kg}\cdot\mathrm{m}$
Torque coefficient	k_τ	$2.8893\cdot10^{-7}$	$\mathrm{kg}\cdot\mathrm{m}^2$

Table I
PHYSICAL QUANTITIES OF THE PROTOTYPE

		k_{ff}	k_p	k_I	k_d	N
Roll/Pitch	outer		2.56			
	inner	0.0266	0.287	0.836	0.007	5.79
Yaw	outer		1.536			
	inner	0.0266	0.287	0.836	0.007	5.79
North/East	outer		8			
	inner	0	4	0.002	0.996	5
Down	outer		8			
	inner	0	4	0.002	0.996	5

Table II
CASCADE P/PID GAINS USED IN THE EXPERIMENTS

was achieved. In particular, the following values have been employed in the IQTO law

$$\begin{gathered} K_R=\mathrm{diag}(0.5,0.5,0.4)\quad K_\omega=\mathrm{diag}(0.35,0.35,0.3)\\ K_I=\mathrm{diag}(0.01,0.01,0.01)\quad K_{\omega_I}=\mathrm{diag}(0.4,0.4,0.4)\\ k_x=(7,7,7)\quad k_v=(7,7,7)\quad k_{I_p}=(7,7,7)\\ k_{I_p}=(7,7,7)\quad M=(10,10,10)\quad M_{I_1}=M_{I_2}=(3,3,3) \end{gathered} \tag{75}$$

and in the FDBL law

$$\begin{gathered} K_{\omega_1}=\mathrm{diag}(0.5,0.5,0.4)\quad K_{\omega_2}=\mathrm{diag}(0.35,0.35,0.3)\\ K_{\omega_3}=\mathrm{diag}(0.5,0.5,0.4)\quad K_{x_1}=\mathrm{diag}(0.35,0.35,0.3)\\ K_{x_2}=\mathrm{diag}(0.5,0.5,0.4)\quad K_{x_3}=\mathrm{diag}(0.35,0.35,0.3). \end{gathered} \tag{76}$$

For what concerns the CPID architecture, the attitude controller parameters have been tuned using a structured $\mathcal{H}_\infty$ approach (Table II, top) which allows one to handle more easily the large number of parameters involved in the control law while optimizing the closed-loop performance. As for the position controller, the parameters obtained with structured $\mathcal{H}_\infty$ have been adjusted (Table II, bottom) to achieve a step response comparable with the other controllers.

C. Stabilization: set-point tracking

The first experiment is intended to show the set-point tracking capabilities of thrust-vectoring UAVs in a scenario in which the UAV has to reach a given location starting from hovering in a misplaced position. This task can be challenging since the tilt-angles and the spinning rates commanded to the actuators are large when the controller is tuned to have high performance and the initial pose errors are large: there may be a significant disturbance torque associated with the control wrench (Remark 4), inducing an undesired coupling between the attitude and position dynamics. We would like to underline that all the experiments that we found in the literature, notably [1], [8], were performed with almost negligible initial pose errors. Throughout the test, a zero-level attitude is also required. Note that this motion could not be performed by coplanar platforms since they can translate only by changing their attitude.

In the experiment, the UAV starts at $x(0)=\begin{bmatrix}0&0&1\end{bmatrix}^\top$ m and has to reach $x_d=\begin{bmatrix}1.2&0&1\end{bmatrix}^\top$ m. The results obtained with the different controllers are reported in Figures 7 for what concerns position tracking errors, while attitude errors, in terms roll-pitch-yaw angle errors, are reported in Figures 8. The norm of the position tracking errors $\|e_x\|$ and the distance on $\mathrm{SO}(3)$ defined by $\|R_e^r\|_{\mathrm{SO}(3)}$ (see the notation section) are reported as well to better highlight the differences among the controllers (Figure 9). The control force and torque computed by the proposed controllers are shown in Figures 10 and the corresponding percentage thrust of each motor and the tilt angles commanded to the servo-actuators are reported in Figures 11. Note that the considered stabilization task is characterized by fast and large inputs sent by the controllers to the actuators: tilt-angles up to 50deg and throttle% up to 80% are instantaneously commanded to the servo-actuators and to the propellers. In this sense such a task is qualified as "aggressive". The FDBL yields the minimum overshoot in terms of position tracking but the CPID architecture achieves the best overall results. Indeed, the response is almost as fast as the FDBL controller but the attitude error is kept much smaller throughout the transient. Note that the aggressive action of the FDBL results in a significant overshoot in terms of attitude angles (see Figure 9 (bottom)): the large control force (see Figure 10 (middle)), which is not exactly applied at the center of mass (Remark 4), together with the required fast tilting of the propellers (see Figure 11 - middle), generate a significant disturbance torque on the attitude dynamics. Although the IQTO provides worse results in position tracking, the control law (68)-(69) guarantees a limited control force which results in a reduced disturbance torque: its attitude performance is basically the same as the one achieved by the CPID controller.

D. Trajectory tracking: combined position and attitude maneuver

The second experiment has the goal of comparing the tracking capabilities of the controllers and, at the same time, of showing the potential of thrust-vectoring UAVs to perform independent attitude and position maneuvers. The desired trajectory is assigned as follows: the platform has to reach a desired pitch angle of 25deg and then it is required to follow a straight segment of 2.5m back and forth, at constant altitude, in 10s, which is assigned as a fifth-order polynomial function of time. The maximum speed and acceleration along the trajectory are $v_M=0.95\mathrm{m/s}$ and $\dot{v}_M=0.6\mathrm{m/s}^2$.

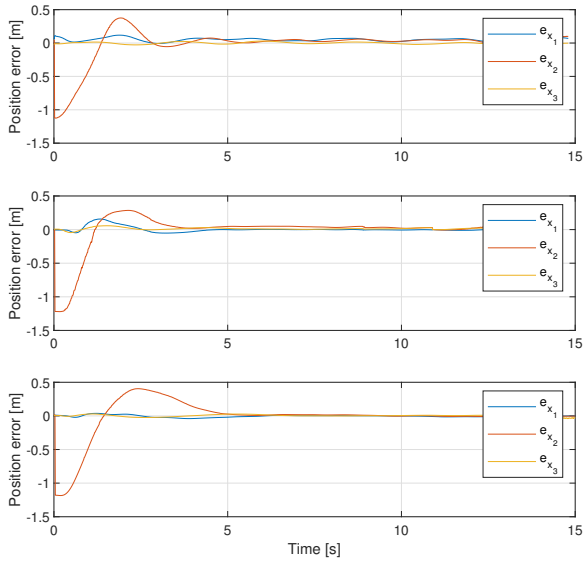

Figure 7. Set-point tracking experiment: position tracking error - e_x - CPID (top), FDBL (middle), IQTO (bottom).

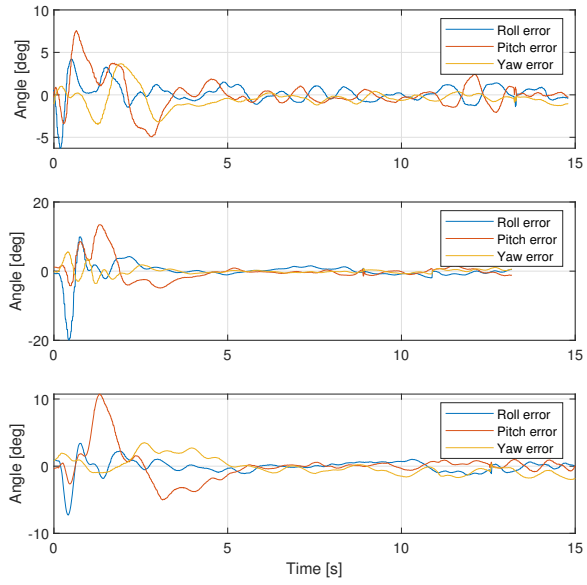

Figure 8. Set-point tracking experiment: attitude error in terms of roll-pitch-yaw angle errors - CPID (top), FDBL (middle), IQTO (bottom).

As done for the first experiment, we collected position tracking errors and the corresponding norm (see Figure 12 and 14 (top)), attitude errors, in terms roll-pitch-yaw angle errors and the distance $\|R_e^r\|_{\mathrm{SO}(3)}$ (see Figure 13 and 14 (bottom)), the control force and torque (Figure 15) and the percentage thrust of each motor and the tilt angles commanded of the servo-actuators (Figure 16), as computed by the different controllers. Table III and IV report the mean and root mean square error (rmse) of the position and attitude tracking errors, respectively

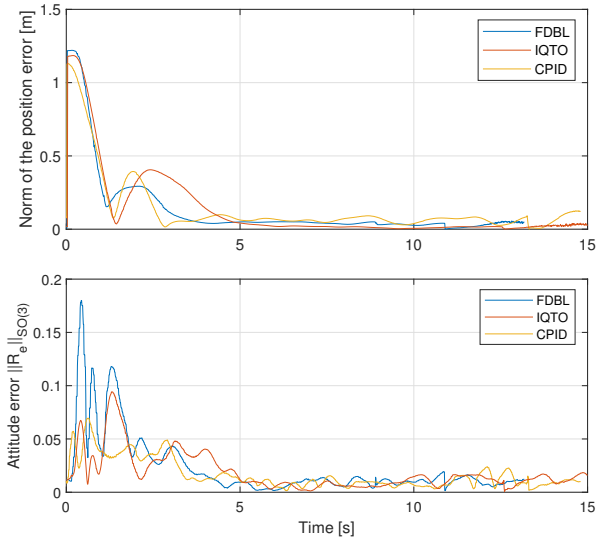

Figure 9. Set-point tracking experiment: comparison of the norm of the position tracking error $\|e_x\|$ (top) - and of the attitude error in terms of normalized distance on $\mathrm{SO}(3)$ - $\|R_e^r\|_{\mathrm{SO}(3)}$ (bottom) - synchronized starting times.

(therein e_ϕ, e_θ, e_ψ represents the error between the desired and actual roll, pitch and yaw angles). By inspecting the results in Table III, it is clear that while FDBL and IQTO yield similar performance, the CPID controller has significantly worse position tracking capabilities. On the other hand, similar performance are achieved by the controllers in terms of attitude errors (see Table IV) for which a constant reference is assigned.

Measure	**CPID**	**FDBL**	**IQTO**
$e_{x_{mean}}$	0.0815	0.0460	0.0023
$e_{y_{mean}}$	0.0681	0.0040	0.0017
$e_{z_{mean}}$	0.0062	0.0134	0.0026
$e_{x_{rms}}$	0.0892	0.0227	0.0117
$e_{y_{rms}}$	0.0992	0.0203	0.0128
$e_{z_{rms}}$	0.0126	0.0213	0.0117

Table III
MEAN AND ROOT MEAN SQUARE ERROR OF THE POSITION TRACKING ERROR COMPONENTS [m]

VII. CONCLUSIONS

In this work we have reviewed several control designs which have been recently proposed for trajectory tracking in UAVs with full actuation capabilities. The considered architectures have been analyzed on theoretical as well as on practical aspects, with specific reference to the trajectory tracking problem. Three algorithms, namely, a quaternion-based P/PID architecture, a feedback linearization controller and a nonlinear tracking controller, have been tuned and tested on a tilt-arm quadrotor UAV. The results of two experiments have been presented: a set-point tracking example and a full-pose

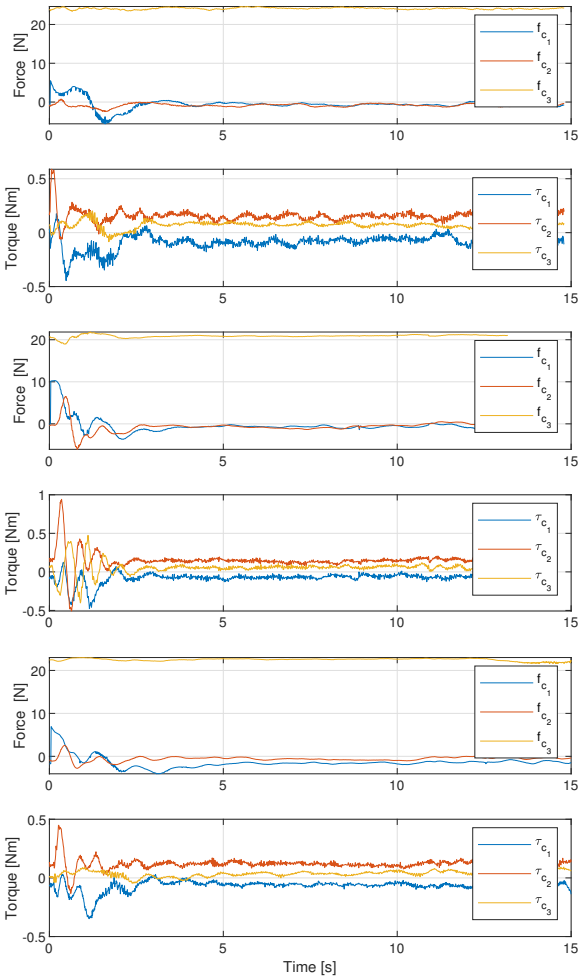

Figure 10. Set-point tracking experiment: control force and torque - (f_c, τ_c) - CPID (top), FDBL (middle), IQTO (bottom).

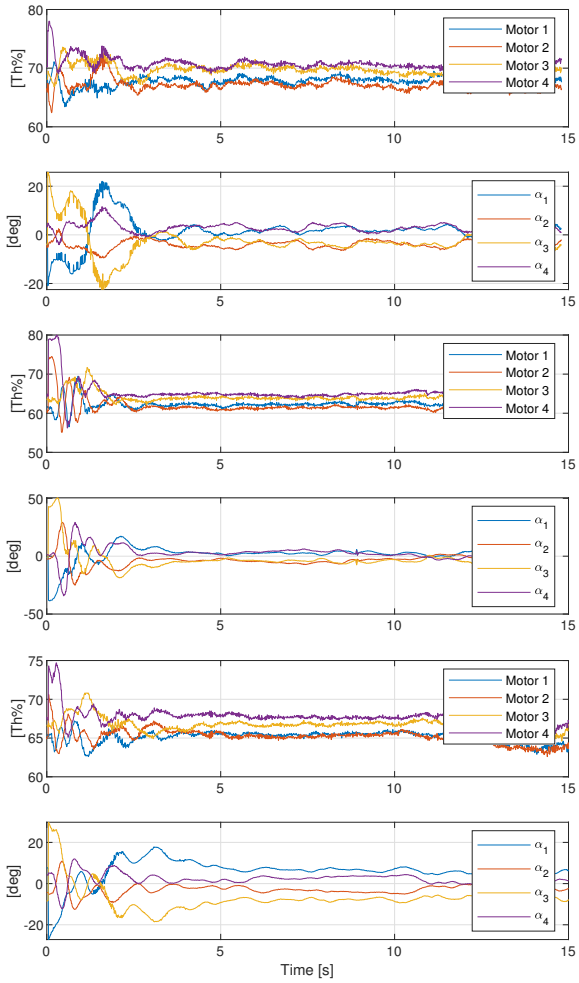

Figure 11. Set-point tracking experiment: Percentage motors thrust and tilt angles - CPID (top), FDBL (middle), IQTO (bottom).

tracking example with time-varying reference. While the CPID achieved the best performance in the set-point tracking task, the FDBL and the IQTO controllers have shown superior performance in the full-pose trajectory tracking scenario. Finally, it is worth mentioning that the IQTO tracking controller, which is based on a quasi-time optimal stabilizer that guarantees a bounded control force by design, is more suitable to handle cases with significant initial errors. This paper is an attempt not only to collect state of the art control designs for the trajectory tracking problem of fully actuated UAVs but also to point out their main limitations and, at the same time, to highlight the main challenges to exploit at best the capabilities of such platforms. To this end we believe that future works should remove some of the most stringiest assumptions outlined in Section III to derive the commonly adopted model for control. In particular, control designs should account for the unavoidable cross-coupling among the position and attitude dynamics (Remark 4) that might seriously affect performance, especially in aggressive tasks.

REFERENCES

[1] M. Ryll, H. H. Bülthoff, and P. Robuffo Giordano, "A novel overactuated quadrotor unmanned aerial vehicle: Modeling, control, and experimental validation," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, pp. 540–556, Mar. 2015.

[2] D. Kastelan, M. Konz, and J. Rudolph, "Fully actuated tricopter with pilot-supporting control," *IFAC-PapersOnLine*, vol. 48, no. 9, pp. 79–84, 2015.

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

[4] B. Crowther, A. Lanzon, M. Maya-Gonzalez, and D. Langkamp, "Kinematic analysis and control design for a nonplanar multirotor vehicle," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 4, pp. 1157–1171, 2011.

[5] S. Rajappa, M. Ryll, H. H. Bülthoff, and A. Franchi, "Modeling, control and design optimization for a fully-actuated hexarotor aerial vehicle with tilted propellers," in *2015 IEEE International Conference on Robotics and Automation*. IEEE, 2015, pp. 4006–4013.

Measure	CPID	FDBL	IQTO
$e_{\phi_{mean}}$	−0.0280	0.1924	−0.5476
$e_{\theta_{mean}}$	0.0097	0.1770	−0.2455
$e_{\psi_{mean}}$	−0.1946	0.0136	−0.5577
$e_{\phi_{rms}}$	1.1630	1.5673	1.7880
$e_{\theta_{rms}}$	1.0400	1.7535	1.9489
$e_{\psi_{rms}}$	1.0996	0.8667	1.6596

Table IV
MEAN AND ROOT MEAN SQUARE ERROR OF THE ROLL, PITCH AND YAW ANGLE ERRORS [deg]

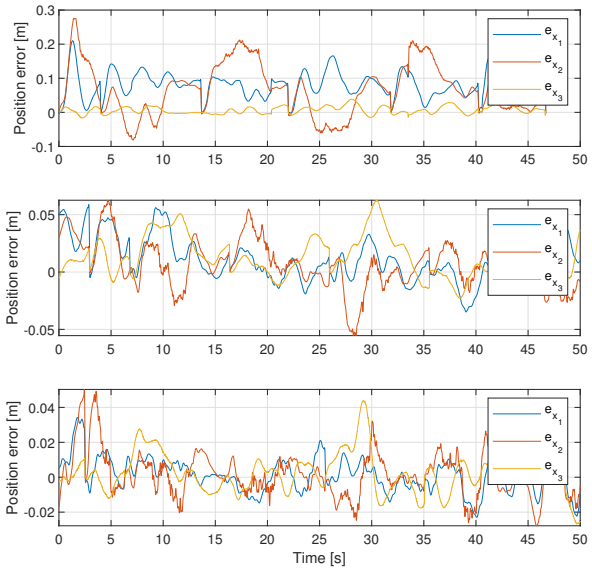

Figure 12. Back and forth motion with 25deg pitch: position tracking error - e_x - CPID (top), FDBL (middle), IQTO (bottom).

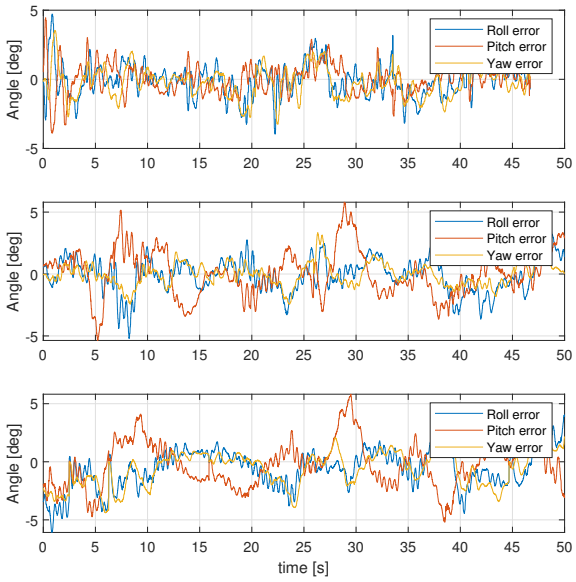

Figure 13. Back and forth motion with 25deg pitch: attitude error in terms of roll-pitch-yaw angle errors - CPID (top), FDBL (middle), IQTO (bottom).

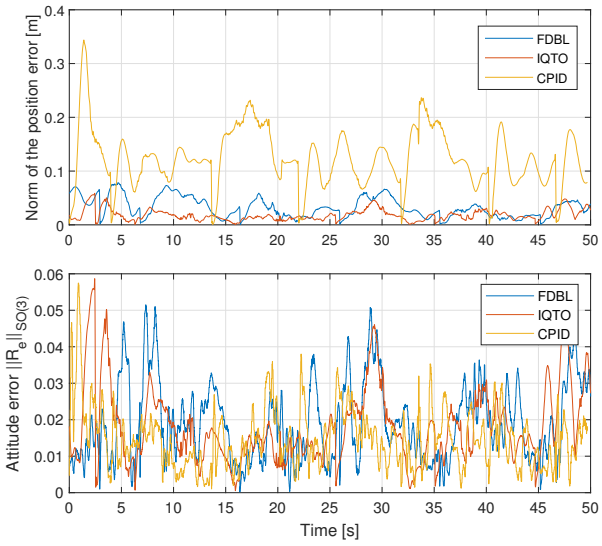

Figure 14. Back and forth motion with 25deg pitch: comparison of the norm of the position tracking error $\|e_x\|$ (top) and of the attitude error in terms of normalized distance on SO(3) $\|R_e^r\|_{SO(3)}$ (bottom) - synchronized starting times.

[6] D. Brescianini and R. D'Andrea, "Design, modeling and control of an omni-directional aerial vehicle," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, May 2016, pp. 3261–3266.

[7] A. Oosedo, S. Abiko, S. Narasaki, A. Kuno, A. Konno, and M. Uchiyama, "Flight control systems of a quad tilt rotor unmanned aerial vehicle for a large attitude change," in *2015 IEEE International Conference on Robotics and Automation*. IEEE, 2015, pp. 2326–2331.

[8] M. Kamel, S. Verling, O. Elkhatib, C. Sprecher, P. Wulkop, Z. J. Taylor, R. Siegwart, and I. Gilitschenski, "The voliro omnidirectional hexacopter: An agile and maneuverable tiltable-rotor aerial vehicle," *IEEE Robotics Automation Magazine*, p. 1, 2018.

[9] A. Franchi, R. Carli, D. Bicego, and M. Ryll, "Full-pose tracking control for aerial robotic systems with laterally bounded input force," *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 534–541, Apr. 2018.

[10] D. Invernizzi and M. Lovera, "Trajectory tracking control of thrust-vectoring UAVs," *Automatica*, vol. 95, pp. 180 – 186, 2018.

[11] M. Furci, C. Nainer, L. Zaccarian, and A. Franchi, "Input allocation for the propeller-based overactuated platform rospo," *IEEE Transactions on Control Systems Technology*, 2019.

[12] D. Invernizzi, M. Lovera, and L. Zaccarian, "Dynamic attitude planning for trajectory tracking in thrust-vectoring UAVs," *IEEE Transactions on Automatic Control*, vol. 65, no. 1, pp. 453–460, Jan. 2020.

[13] PX4-Community, "Documentation available at https://docs.px4.io/en/," Tech. Rep., 2018.

[14] T. Fernando, J. Chandiramani, T. Lee, and H. Gutierrezee, "Robust adaptive geometric tracking controls on SO(3) with an application to the attitude dynamics of a quadrotor UAV," in *50th IEEE Conference on Decision and Control and European Control Conference*, Dec. 2011, pp. 7380–7385.

[15] D. Brescianini, M. Hehn, and R. D'Andrea, "Nonlinear quadrocopter attitude control," *Department of Mechanical and Process Engineering, ETHZ, Tech. Rep*, 2013.

[16] F. Goodarzi, D. Lee, and T. Lee, "Geometric nonlinear PID control of a quadrotor UAV on se(3)," in *Proc. European Control Conf. (ECC)*, Jul. 2013, pp. 3845–3850.

[17] F. Forni, S. Galeani, and L. Zaccarian, "A family of global stabilizers for quasi-optimal control of planar linear saturated systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 5, pp. 1175–1180, May

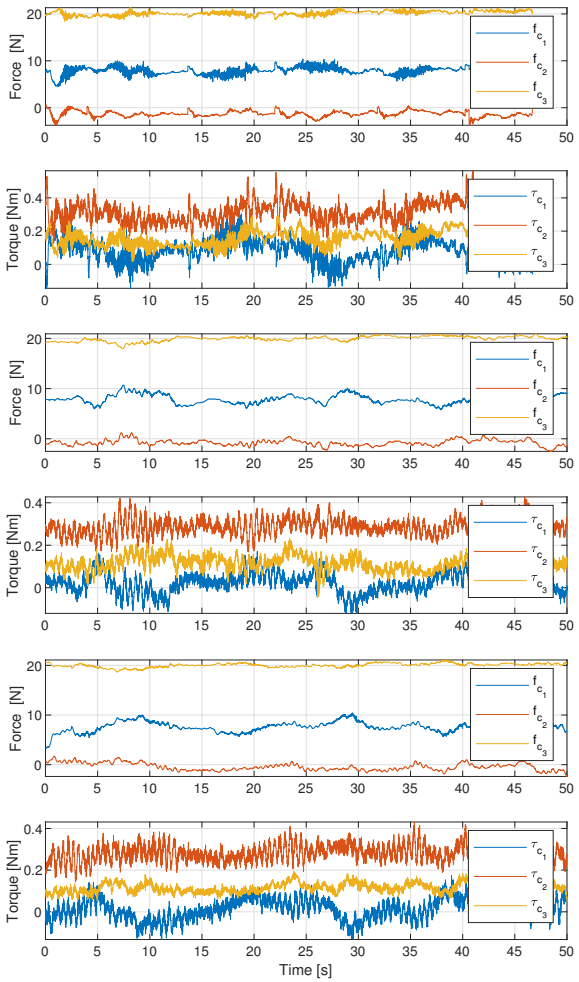

Figure 15. Back and forth motion with 25deg pitch: control force and torque - (f_c, τ_c) - CPID (top), FDBL (middle), IQTO (bottom).

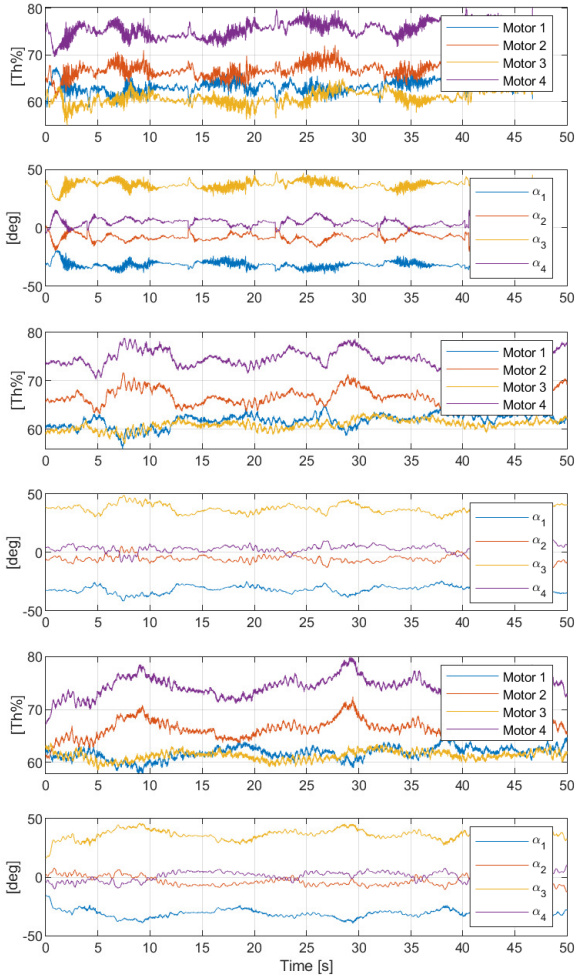

Figure 16. Back and forth motion with 25deg pitch: Percentage motors thrust and tilt angles - CPID (top), FDBL (middle), IQTO (bottom).

2010.

[18] M.-D. Hua and C. Samson, "Time sub-optimal nonlinear PI and PID controllers applied to longitudinal headway car control," *International Journal of Control*, vol. 84, no. 10, pp. 1717–1728, 2011.

[19] Riccardi, F. and Lovera, M., "Robust attitude control for a variable-pitch quadrotor." in *IEEE Conference on Control Applications, Antibes, France*, 2014, pp. 730–735.

[20] J. Shen, "Nonlinear control of multibody systems with symmetries via shape change," Ph.D. dissertation, University of Michigan, 2002.

[21] F. Bullo and M. R. M., "Tracking for fully actuated mechanical systems: a geometric framework," *Automatica*, vol. 35, no. 1, pp. 17–34, jan 1999.

[22] Mahony, R., Kumar, V. and Corke, P., "Multirotor Aerial Vehicles: Modeling, Estimation and Control of Quadrotor," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[23] D. Mellinger, N. Michael, and V. Kumar, "Trajectory generation and control for precise aggressive maneuvers with quadrotors," in *Experimental robotics*. Springer, 2014, pp. 361–373.

[24] D. Falanga, E. Mueggler, M. Faessler, and D. Scaramuzza, "Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 5774–5781.

[25] F. Bullo and A. D. Lewis, *Geometric Control of Mechanical Systems: Modeling, Analysis, and Design for Mechanical Control Systems*. Springer, 2005.

[26] D. E. Koditschek, "The Application of Total Energy as a Lyapunov Function for Mechanical Control Systems," *J. E. Marsden, P. S. Krishnaprasad and J. C. Simo (Eds) Dynamics and Control of Multi Body Systems*, vol. 97, pp. 131–157, February 1989.

[27] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel, "Quaternion-based hybrid control for robust global attitude tracking," *IEEE Transactions on Automatic Control*, vol. 56, no. 11, pp. 2555–2566, Nov 2011.

[28] E. Kaufman, K. Caldwell, D. Lee, and T. Lee, "Design and development of a free-floating hexrotor UAV for 6-dof maneuvers," in *2014 IEEE Aerospace Conference*. IEEE, 2014, pp. 1–10.

[29] C. Micheli, "Design, identification and control of a tiltrotor quadcopter UAV," Master's thesis, Politecnico di Milano, 2016.

[30] M. Giurato, "Design, integration and control of a multirotor UAV platform," Master's thesis, Politecnico di Milano, 2015.