D. Invernizzi, S. Panza, M. Lovera
Robust Tuning of Geometric Attitude Controllers for Multirotor Unmanned Aerial Vehicles
Journal of Guidance Control and Dynamics, Vol. 43, N. 7, 2020, p. 1332-1343
doi:10.2514/1.G004457

Robust tuning of geometric attitude controllers for multirotor Unmanned Aerial Vehicles

Davide Invernizzi *, Simone Panza †, and Marco Lovera ‡
Politecnico di Milano, Milan, Italy, 20156

In recent years there has been a significant body of literature proposing nonlinear attitude control laws for small-scale multirotor UAVs, motivated by the high maneuverability of these platforms. While tracking trajectories characterized by fast and large attitude changes makes the control problem intrinsically nonlinear, most of the works proposing nonlinear designs is concerned with establishing their stabilizing properties, often deduced by referring to simplified dynamical models, but limited attention has been devoted to performance. As a consequence, less satisfactory results than expected are typically achieved in experiments and the controller gains must be adjusted with trial and error procedures to obtain good performance. In this paper we propose a model-based tuning method that exploits the cascade structure of the attitude dynamics and that needs only single-axis identified linear models of the angular velocity dynamics to be applied. The tuning of the controller gains is carried out on the linearized closed-loop system with structured H_∞ synthesis which allows one to enforce robustness against model uncertainty in a systematic way and to achieve a desired level of performance in nominal conditions. The approach is validated by tuning the gains of a novel P/PID-like cascade, which has been developed in the framework of geometric control theory. A thorough analytical comparison of the proposed design with a geometric PI-like controller borrowed from the literature is complemented with experiments conducted on a small quadrotor UAV.

Nomenclature

O_I	=	origin of the inertial frame.
O_B	=	origin of the body frame attached to the UAV.
$b_i \in \mathbb{S}^2$	=	*i-th* unit vector of the body frame.
$e_i \in \mathbb{S}^2$	=	*i-th* unit vector of the inertial frame.
$x \in \mathbb{R}^3$	=	position of the UAV body frame with respect to O_I resolved in the inertial frame [m].
$R \in \mathrm{SO}(3)$	=	rotation matrix describing the attitude of the UAV.

*Assistant Professor, Department of Aerospace Science and Technology, via La Masa, 34.
†Post-doc Research Assistant, Department of Aerospace Science and Technology, via La Masa, 34.
‡Full Professor, Department of Aerospace Science and Technology, via La Masa, 34.

$v \in \mathbb{R}^3$ = velocity of the UAV at O_B [m/s].

$\omega \in \mathbb{R}^3$ = body angular velocity of the UAV.

$\tau_c \in \mathbb{R}^3$ = torque at point O_B associated with the propellers action [Nm].

$\tau_e \in \mathbb{R}^3$ = torque at point O_B associated with exogenous actions [Nm].

$f_{p_i} \in \mathbb{R}^3$ = force delivered by the *i-th* propeller [N].

$\tau_{p_i} \in \mathbb{R}^3$ = torque delivered by the *i-th* propeller at the *i-th* rotor hub [Nm].

$T_i \in \mathbb{R}_{\geq 0}$ = magnitude of f_{p_i} [N].

$\omega_{r_i} \in \mathbb{R}_{>0}$ = angular rate of the *i-th* propeller [rad/s].

$\dot{\omega}_{r_i} \in \mathbb{R}$ = angular acceleration of the *i-th* propeller [rad/s^2].

$\varepsilon_i \in \{-1, 1\}$ = rotation direction of the *i-th* rotor (positive counterclockwise).

$k_f \in \mathbb{R}_{>0}$ = thrust coefficient of the rotors [kgm].

$k_\tau \in \mathbb{R}_{>0}$ = torque coefficient of the rotors [kgm^2].

$\sigma \in \mathbb{R}_{>0}$ = ratio between k_f and k_τ.

$x_{br_i} \in \mathbb{R}^3$ = position vector from the airframe origin to the hub of the *i-th* rotor disk [m].

$\gamma_i \in [0, 2\pi)$ = angle of the between b_1 and the *i-th* arm in the counterclockwise direction [rad]

$\ell \in \mathbb{R}_{>0}$ = distance between the UAV center of mass and the rotors hubs [m]

$J \in \mathbb{R}^{3\times3}$ = inertia matrix resolved in the body frame [kgm^2]

$J_{r_i} \in \mathbb{R}^{3\times3}$ = inertia matrix of the *i-th* rotor resolved in the rotor frame [kgm^2]

$S \in \mathbb{R}^{3\times4}$ = First moment of inertia resolved in the body frame [kgm]

$m \in \mathbb{R}_{>0}$ = mass of the UAV [kg]

I. Introduction

Attitude control for multirotor Unmanned Aerial Vehicles (UAVs) is of fundamental importance since their flying qualities depend significantly on the performance and stabilizing properties of their attitude controllers, the design and tuning of which must be carried out in a sensible way. Guaranteeing good attitude tracking performance is fundamental to cope with the underactuated nature of UAVs with coplanar propellers, in which the attitude dynamics is used to stabilize the position dynamics. Depending on the application, requirements may vary and different modeling and control law design tools have to be considered. If one is concerned with applications such as inspection, surveillance, mapping, video and photography then linear modeling and control design methods are suitable and allow handling stringent performance requirements in a systematic way. On the other hand, when considering maneuvers involving significant changes in attitude such as in perching operations on inverted surfaces with high precision and repeatability, in passing through narrow, vertical gaps or in fast flight conditions with dynamic obstacle avoidance, linear controllers

may have deficiencies and yield poor performance or, even worse, fail to stabilize the vehicle.

While model-based or data-driven approaches have been presented to tune linear attitude controllers for multirotor UAVs [1, 2], for nonlinear control laws no systematic approaches exist, to the best of our knowledge. Global tracking properties of such control laws are usually derived by treating multirotor UAVs as rigid bodies [3–6] while robust nonlinear designs are typically limited to the case of uncertain inertia matrix [7] and constant exogenous disturbances [8]. However, the rigid body model fails to be representative of the actual dynamics of a multirotor UAV, which includes the motor dynamics, aerodynamic effects, measurements noise and delays, as can be seen by inspecting identified models of the attitude dynamics in near hovering conditions [9]. It is therefore inevitable that the tuning phase is challenging: starting from initial values obtained on a simulation platform, gains are adjusted with trial and error procedures on the real platform [8]. Clearly, to achieve satisfactory performance and to save time, the simulation platform must be sufficiently accurate. This, in turn, asks for the identification of several parameters, an activity which is not only time-consuming but prone to error as well. Even when a reasonably accurate simulation model is available, the tuning procedure on the simulation platform is still based on trial and error and therefore sensitive to the control designer experience. An alternative solution to deal with such issue is to develop nonlinear adaptive schemes which, nonetheless, results in a more complex control law [10–12].

On the contrary, the approach presented in this work is an attempt to simplify the tuning procedure of nonlinear control laws for attitude control of small-scale multirotor UAVs by exploiting the cascade structure of the attitude motion, in which nonlinearities are mostly associated with the kinematics while the main source of uncertainties is the angular velocity dynamics. In particular, the proposed tuning method requires only the identification of single-axis linear models of the angular velocity dynamics. This is motivated by the fact that gyroscopic terms have a limited influence on the dynamics of small-scale UAVs and together with other nonlinear effects they can be treated as disturbances to be rejected by the control system. Identified models, although linear and decoupled, allow capturing more accurately the angular velocity dynamics which includes the motor dynamics as well as aerodynamic effects. By referring to the linearized closed-loop system obtained with the identified models, robust tuning techniques can be employed to tune controller gains. In this work, structured H_∞ [13] has been employed since it was proved to be of great practical usefulness and has been widely adopted in aerospace applications, in particular on rotorcraft [14–16]. Specifically, it may be used to enforce robustness against model uncertainty and requirements encoded in the frequency domain can be used to achieve tight performance. By referring to identified models of a small quadrotor UAV, the proposed approach has been validated by tuning the gains of a novel P/PID-like cascade architecture, which has been developed in the framework of geometric control theory (see [17] for an introduction to geometric control). A PI-like geometric controller inspired by [18] has been considered as well for comparison purposes. Both controllers are globally well-defined, simple to implement and, thanks to the robust tuning procedure, with guaranteed performance in near hovering conditions. This claim has been further verified by conducting experiments on a small quadrotor UAV: the tests confirmed that the

behavior of the control architectures is consistent with the closed-loop sensitivity functions obtained with the identified model. Furthermore, the controllers performed with satisfactory results also when large amplitude references were required. While the cascade controller showed excellent results in aggressive set-point tracking, when large and constant attitude changes are commanded, the PI-like controller provided good trajectory tracking performance.

There are three main contributions in this work, which can be summarized as follows. Firstly, a novel nonlinear cascade architecture, capable of high performance, has been presented and its stability and tracking capabilities have been studied and discussed in details. Secondly, a systematic approach to tune the gains of nonlinear attitude control architectures has been presented. The approach requires only single-axis identified models of the angular velocity dynamics and allows recovering high performance in near hovering conditions typical of linear control designs while guaranteeing, by design, desirable tracking properties in a broader range of operations. Finally, the proposed analysis, complemented with experimental results, has allowed showing strong points and deficiencies of the considered architectures and their suitability in relation to the considered applications.

II. Notation

Notation. In this paper $\mathbb{Z}$ denotes the set of integers, $\mathbb{R}$ ($\mathbb{R}_{>0}, \mathbb{R}_{\geq 0}$) 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. With a slight abuse of notation we use the same notation for a vector $x \in \mathbb{R}^n$ and its representation as column matrix in some basis. The canonical basis in $\mathbb{R}^n$ is denoted as $e_i := (0, \ldots, 1, \ldots, 0)$ for $i \in \{1, ..., n\}$ and the identity matrix in $\mathbb{R}^{n\times n}$ is denoted as $I_n := [e_1 \cdots e_i \cdots e_n]$. Given $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$, $\|x\| := \sqrt{x_1^2 + \ldots + x_n^2}$ is the corresponding Euclidean norm. Given $A \in \mathbb{R}^{n\times n}$, we use the compact notation $A \in \mathbb{R}^{n\times n}_{>0}$ ($\mathbb{R}^{n\times n}_{<0}$) to represent a positive (negative) definite matrix. The set $\mathrm{SO}(3) := \{R \in \mathbb{R}^{3\times 3} : R^T R = I_3, \det(R) = 1\}$ denotes the third-order Special Orthogonal group while $\mathbb{S}^n := \{q \in \mathbb{R}^{n+1} : \|q\| = 1\}$ denotes the n-dimensional unit sphere. The *hat* map $\hat{\cdot} : \mathbb{R}^3 \to \mathfrak{so}(3)$, given by

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

for any $\omega := (\omega_1, \omega_2, \omega_3) \in \mathbb{R}^3$, defines an isomorphism between $\mathbb{R}^3$ and the vector space of third-order skew-symmetric matrices, *i.e*, $\mathfrak{so}(3) := \{W \in \mathbb{R}^{3\times 3} : W = -W^T\}$ with the corresponding inverse given by the *vee* map $(\cdot)^\vee : \mathfrak{so}(3) \to \mathbb{R}^3$. For $\hat{\omega} \in \mathfrak{so}(3)$, one has $\hat{\omega} y = \omega \times y$, $\forall y \in \mathbb{R}^3$, where $\times$ is the cross product. The notation $R_u(\theta) := \exp(\theta \hat{u})$ is used to represent the rotation matrix corresponding to a rotation about a unit axis u of an angle $\theta \in \mathbb{R}$.

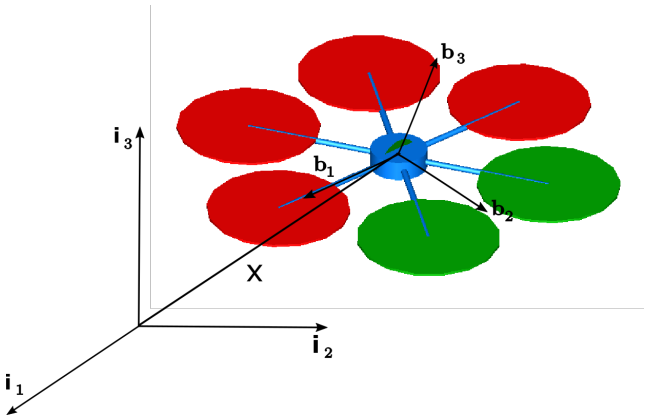

Fig. 1 Example of a multirotor UAV and definition of the frames.

III. The attitude dynamical model of multirotor UAVs

A multirotor UAV is an aerial vehicle made by a central body and n arms, each of which carries a propeller group. The propeller groups consist of a motor and a propeller and are in charge of producing the wrench (force and torque) required to control the motion of the UAV (Figure 1). The configuration of the UAV can be identified with the motion of a body-fixed frame $\mathcal{F}_B := (O_B, \{b_1, b_2, b_3\})$ with respect to a reference frame $\mathcal{F}_I := (O_I, \{i_1, i_2, i_3\})$, where b_j and i_j ($j \in \{1, 2, 3\}$) are unit vectors forming right-handed orthogonal triads and O_B, O_I are the origins of the body and reference frame, respectively. The position vector from O_I to O_B, resolved in $\mathcal{F}_I$, is denoted hereafter as $x \in \mathbb{R}^3$. We denote $R := [\, b_1 \; b_2 \; b_3 \,] \in \mathrm{SO}(3)$ the rotation matrix describing the attitude of the UAV, where b_i are the body axes resolved in $\mathcal{F}_I$.

By referring to the body frame, the attitude motion is described by the following set of equations:

$$\dot{R} = R\hat{\omega} \tag{2}$$

$$J\dot{\omega} = -\omega \times \left(J\omega + \sum_{i=1}^{n} J_{r_i} e_3 \dot{\omega}_{r_i} \right) + \sum_{i=1}^{n} J_{r_i} e_3 \dot{\omega}_{r_i} + S\dot{v} + \tau_e(R, x, \omega, v, g, t) + \tau_c \tag{3}$$

where $J = J^T \in \mathbb{R}^{3\times3}_{>0}$ is the inertia matrix with respect to O_B, $S \in \mathbb{R}^{3\times3}$ is the first moment of inertia, $J_{r_i} = J_{r_i}^T \in \mathbb{R}^{3\times3}_{>0}$ such that $R_{e_3}(\alpha) J_{r_i} R_{e_3}^T(\alpha) = J_{r_i} \; \forall \alpha \in \mathbb{R}$ is the inertia matrix of the *i-th* propeller, $\omega \in \mathbb{R}^3$ is the body angular velocity, $v \in \mathbb{R}^3$ is the translational velocity, $\tau_c, \tau_e \in \mathbb{R}^3$ are the control and disturbance torque, respectively, and $\omega_r := (\omega_{r_1}, \ldots, \omega_{r_n}) \in \mathbb{R}_{\geq 0} \times \ldots \times \mathbb{R}_{\geq 0}$ contains the angular rates of the propellers. The control torque τ_c delivered by the propellers at O_B is given by:

$$\tau_c := \sum_{i=1}^{n} x_{br_i} \times f_{p_i} + \tau_{p_i} \tag{4}$$

where f_{p_i} and $\tau_{p_i} \in \mathbb{R}^3$ are, respectively, the force and torque delivered by the *i-th* rotor and $x_{br_i} \in \mathbb{R}^3$ is the position

vector from O_B to the hub of the *i-th* rotor disk, all resolved in $\mathcal{F}_B$. In this work we rely on the widely adopted quadratic aerodynamic model for the propeller force and torque generation mechanism [19]. According to this model, the components in the propeller frame of the force and torque delivered by *i-th* propeller are:

$$f_{p_i} := k_f \omega_{r_i}^2 e_3 \tag{5}$$

$$\tau_{p_i} := -\varepsilon_i k_\tau \omega_{r_i}^2 e_3, \tag{6}$$

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 of the *i-th* rotor. Note that this model is valid when considering small deviations from the hovering condition [19] although it has been successfully used in control laws for highly acrobatic maneuvering, as documented by experimental works [20, 21]. By defining

$$T_i := k_f \omega_{r_i}^2 \qquad i = 1, \ldots, n, \tag{7}$$

the *i-th* propeller force and torque can be written as $f_{p_i} = T_i e_3$ and $\tau_{p_i} = -\varepsilon_i \sigma T_i e_3$, respectively, where $\sigma := k_\tau / k_f$ is a positive constant. Then, when assuming that there are sufficiently fast low-level controllers to track any desired angular rate ω_{r_i}, $u := (T_1, \ldots, T_n) \in \mathbb{R}_{\geq 0} \times \ldots \times \mathbb{R}_{\geq 0}$ can be considered as the input variable for control design. If the rotor hubs are placed equidistantly from the center of mass, each making an angle $\gamma_i \in [0, 2\pi)$ with respect to b_1, the control torque (4) can be written as a map $(T_1, \ldots, T_n) \mapsto \tau_c(T_1, \ldots, T_n)$, given by the following expression:

$$\tau_c(T_1, \ldots, T_n) := \sum_{i=1}^{n} \left(x_{br_i} \times T_i e_3 - \varepsilon_i \sigma T_i e_3 \right) \tag{8}$$

where $x_{br_i} := \ell R_{e_3}(\gamma_i) e_1$ with ℓ being the distance between the rotor hubs and O_B. Finally, the total force is delivered by the propellers along the positive direction of the third body axis, according to the following map:

$$f_c(T_1, \ldots, T_n) := \sum_{i=1}^{n} T_i e_3. \tag{9}$$

Note that T_i appears linearly in (8)-(9). Hence, by defining $T_c := f_c^T e_3$, the input mapping for a multirotor UAV can be

compactly written in matrix form as:

$$\begin{bmatrix} T_c \\ \tau_{c_1} \\ \tau_{c_2} \\ \tau_{c_3} \end{bmatrix} = \begin{bmatrix} 1 & \cdots & 1 \\ \ell \sin(\gamma_1) & \cdots & \ell \sin(\gamma_n) \\ -\ell \cos(\gamma_1) & \cdots & -\ell \cos(\gamma_n) \\ -\varepsilon_1 \sigma & \cdots & -\varepsilon_n \sigma \end{bmatrix} \begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix} =: \begin{bmatrix} M_f \\ M_\tau \end{bmatrix} \begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix} \tag{10}$$

where $M_f \in \mathbb{R}^{1\times n}$ and $M_\tau \in \mathbb{R}^{3\times n}$ are the force and torque input mapping, respectively. It is worth remarking that since we have employed an approximated model in (5)-(6), an additional disturbance torque $\Delta\tau_e(u) := \sum_{i=1}^{n} x_{br_i} \times \Delta f_{p_i}(u) + \Delta\tau_{p_i}(u)$ should be included in equation (3) to account for modeling errors. The remaining part of the external torque is related to the aerodynamic interaction of the UAV main body and arms with air and gravity terms, not reported here for the sake of conciseness. Therefore, a general model for the attitude dynamics would be described by:

$$J\dot{\omega} = -\omega \times \left(J\omega + \sum_{i=1}^{n} J_{r_i} e_3 \dot{\omega}_{r_i} \right) + \sum_{i=1}^{n} J_{r_i} e_3 \dot{\omega}_{r_i} + S\dot{v} + \tau_e(R, x, \omega, v, g, t) + M_\tau u + \Delta\tau_e(u). \tag{11}$$

Note that ω_r and $\dot{\omega}_r$ depend upon input u and that in general, the exact expression of τ_e is difficult to find. The model gets even more complicated when one considers the dynamics of the motors which are usually considered as saturated first order low-pass filters with time constant τ_m and time delay t_d:

$$u = \text{sat}_{T_m}^{T_M} \left(\frac{1}{\tau_n s + 1} e^{-s t_d} u_d \right) = \text{sat}_{T_m}^{T_M} \left(G_m(s) u_d \right), \tag{12}$$

where $\text{sat}_{T_m}^{T_M}(x) := \min(\max(T_m, x), T_M)$ is the standard saturation function applied component-wise to x and $T_M \gg T_m \geq 0$ are the upper and lower saturation levels on the propeller thrust.

IV. The attitude tracking problem

As can be seen from equation (10), the coplanar propellers UAV is an underactuated mechanical system: no force can be instantaneously delivered in the plane spanned by b_1, b_2. To control position, the standard approach assumes full actuation of the attitude subsystem to stabilize the position error dynamics. In practice, a control law for τ_c is designed to tilt the body vector b_3 in the direction of the force required for position tracking. At the same time, the magnitude of the control force T_c is adjusted to match the magnitude of the required force and then the propellers thrust (input u) is computed by inverting (10). Therefore, the design of control laws for attitude tracking becomes a fundamental ingredient to ensure the stability of the overall system. If one is not interested in position tracking, the fully actuated rotational dynamics can be exploited to perform arbitrary rotational maneuvers.

As shown in the previous section, the attitude dynamics of multirotor UAVs is fairly complex and is coupled with the translational dynamics. For control design purposes, some simplifying assumptions are considered. In particular, it is often [19] assumed that the origin O_B of the body-fixed frame $\mathcal{F}_B$ coincides with the center of mass of the UAV and that the body axes b_i are principal ones. Furthermore, the inertial and gyroscopic terms related to the spinning of the rotors are assumed to be negligible (small rotor inertia J_{r_i}) together with aerodynamic effects and $\Delta\tau_e(u)$. Under these assumptions, the attitude dynamics reduces to

$$J\dot{\omega} = -\omega \times J\omega + M_\tau \mathrm{sat}_{T_m}^{T_M} \left(G_m(s)u_d\right). \tag{13}$$

Since M is full row rank by design, one can exploit the allocation $u_d = M^+ \begin{bmatrix} T_c^d \\ \tau_c^d \end{bmatrix}$, where $u_d := (T_1^d, \ldots, T_n^d)$ collects the thrusts corresponding to desired T_c^d, τ_c^d to be provided by the controller and $M^+ = M^T (MM^T)^{-1}$ is the right inverse of M, to obtain:

$$J\dot{\omega} = -\omega \times J\omega + M_\tau \mathrm{sat}_{T_m}^{T_M} \left(G_m(s)M^+ \begin{bmatrix} T_c^d \\ \tau_c^d \end{bmatrix}\right). \tag{14}$$

In most operating conditions, it is possible to obtain feasible thrusts u_d (*i.e.*, within the bounds defined by (12)) for some bounded torque and thrust according to the (right) pseudo inverse of (10). In such a case, since $M_\tau \mathrm{sat}_{T_m}^{T_M} \left(G_m(s)M^+ \begin{bmatrix} T_c^d \\ \tau_c^d \end{bmatrix}\right) = G_m(s)M_\tau M^+ \begin{bmatrix} T_c^d \\ \tau_c^d \end{bmatrix} = G_m(s)\tau_c^d$, the simplified model becomes

$$J\dot{\omega} = -\omega \times J\omega + G_m(s)\tau_c^d \tag{15}$$

where τ_c^d is the input variable to be assigned by control design.

Remark 1 *Most of the works in the literature assume the motor dynamics be sufficiently faster than the expected attitude dynamics and consider* $G_m(s) = 1$ *for control design purposes: according to this assumption, the attitude dynamics of a multirotor would be the one of a fully actuated rigid body.*

Remark 2 *For small scale UAVs, the gyroscopic coupling* $\omega \times J\omega$ *is small compared to the other terms and can be reasonably neglected. The approximated attitude dynamics is described in this case by three independent equations of the form:*

$$\omega_i = \frac{G_m(s)}{J_i s}\tau_{c_i}^d =: G_{\omega_i}(s)\tau_{c_i}^d. \tag{16}$$

The performance of control designs based on this simplified model heavily depends upon the knowledge of the principal moments of inertia, which can be either obtained from a CAD model or estimated via identification experiments. Note, in passing, that the inertia moments J_i *are nothing but scaling terms of the control variables* $\tau_{c_i}^d$ *and "wrong" values of* J_i *can be compensated by tuning the gains of the control law to be developed for* τ_c^d*. The outcomes of this (tedious) approach are strongly related to the experience of the control designer: accepting the decoupled axes approximation,*

it is much more convenient to work directly with identified linear models $G_{\omega_i}(s)$ from $\tau_{c_i}^d$ to ω_i for $i = 1, 2, 3$, which provide significant information about the overall system dynamics.

Remark 3 *In deriving the simplified equations of motion, one point of concern is related to the assumption O_B be coincident with the center of mass. Indeed, if this was not correct, one should consider that the effect of gravity does not disappear from τ_e, namely, that a torque $\tau_e^d := SgR^T e_3$ would act as perturbation on the attitude dynamics. Such a disturbance together with $\Delta\tau_e(u)$ (which, among other things, includes terms related to the thrust axes being not exactly aligned with b_3) could significantly affect the UAV flying capabilities. For instance, when referring to near hovering conditions, an almost constant disturbance torque Sge_3 would perturb the attitude dynamics. If not compensated, such a disturbance would make the UAV drift in position due to its underactuated dynamics, with possible destructive consequences.*

Under these premises, the equations of motion that will be used for control design are equation (2) and

$$J\dot{\omega} = -\omega \times J\omega + \tau_c^d + \tau_e, \tag{17}$$

in which $\tau_e \in \mathbb{R}^3$ is assumed to be a constant disturbance torque. Then, attitude tracking problem can be formalized as follows.

Problem 1 *Consider the attitude dynamics described by equations* (2)*,* (17)*. Given a trajectory $t \mapsto (R_d(t), \omega_d(t)) \in \mathrm{SO}(3) \times \mathbb{R}^3$, where $\omega_d(t) := (R_d^T(t)\dot{R}_d(t))^\vee$ is a continuously differentiable and bounded function of time with bounded time derivative, find a control torque τ_c^d such that the trajectory (R_d, ω_d) is locally asymptotically tracked when assuming that full state information is available.*

In this work we first present a cascade control law to tackle the attitude tracking problem by accounting for nonlinearities as well as disturbances. Then, we review a slightly modified version of the nonlinear PI-like control law proposed in [18] which is considered for comparison purposes. Both control laws are geometric in the sense that they are not based on any parametrizations of SO(3).

V. Geometric P/PID-like cascade architecture

The cascade strategy that we propose is motivated by the structure of equations (2), (11) when treating τ_e as an exogenous disturbance. Under this assumption, the attitude dynamics (11) is independent from the kinematics (2). Therefore, the angular velocity ω can be considered as a virtual input to track the desired attitude $t \mapsto R_d(t)$. Then, the torque τ_c^d can be designed according to different control strategies to track the virtual angular velocity, relying on the full actuation assumption of the attitude dynamics.

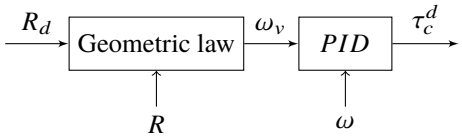

Fig. 2 Cascade attitude control architecture

A. Control law design

We propose the following geometric-based cascade control law (see also Figure 2):

$$\omega_v := -e_R(R_e) + R_e^T \omega_d \tag{18}$$

$$\tau_c^d := PI(s)\,(\omega_v - \omega) - D(s)\omega \tag{19}$$

where $R_e := R_d^T R \in \mathrm{SO}(3)$,

$$e_R := \tfrac{1}{2}\left(K_R R_e - R_e^T K_R\right)^\vee, \tag{20}$$

and $PI(s) := K_p + K_i \frac{1}{s}$, $D(s) := K_d \frac{s}{1+\frac{s}{N}}$ are transfer functions on the Laplace domain defining, respectively, a proportional-integral and (filtered) derivative action. Herein, K_R, K_p, K_i and $K_d \in \mathbb{R}^{3\times3}$ are positive definite diagonal matrices while $N \in \mathbb{R}_{>0}$ is the derivative filter constant; K_{x_i} refers to the i-th element on the diagonal of the generic matrix K_x. Note that the derivative action $D(s)$ is applied only to the state variable and not to the error so that step-like references from the outer loop do not result in excessive peaks in the commanded torque (19).

Theorem 1 *Consider the attitude motion described by equations* (2)*,* (17) *and a desired trajectory* $t \mapsto (R_d(t), \omega_d(t)) \in \mathrm{SO}(3) \times \mathbb{R}^3$*, where* $\omega_d(t) = 0$*. There exist diagonal gain matrices* K_R*,* K_p*,* K_i *and* $K_d \in \mathbb{R}^{3\times3}_{>0}$ *such that the equilibrium point* $(R, \omega, e_I) = (R_d, 0, K_i^{-1}\tau_e)$*, where* $e_I \in \mathbb{R}^3$ *is the state of the integrator, is locally asymptotically stable.*

Proof of Theorem 1. We start the proof by deriving the error dynamics associated with the control law (18)-(19):

$$\dot{R}_e = R_e \hat{\omega} \tag{21}$$

$$J\dot{\omega} = -\omega \times J\omega - PI(s)\omega_v - PID(s)\omega + \tau_e, \tag{22}$$

where $PID(s) := PI(s) + D(s)$ (for simplicity we assume the derivative term $D(s) = K_d s$ in the stability analysis), for which the set of equilibria is given by $\left\{(R_e, \omega, e_I) \in \mathrm{SO}(3) \times \mathbb{R}^3 \times \mathbb{R}^3 : \omega_v = e_R(R_e) = 0,\ \omega = 0,\ e_I = K_i^{-1}\tau_e\right\}$. Such a set contains the desired equilibrium $R_e = I_3$ and three other points in SO(3), namely, $R_e = R_{e_i}(k\pi_i)$ where $k \in \mathbb{Z}$ and $\forall i \in \{1, 2, 3\}$ [23]. To prove local stability of the desired equilibrium, consider the linearized dynamics for

small attitude errors $R_e \approx I_3 + \hat{\theta}_e$, $\omega \approx 0$, where $\theta_e \in \mathbb{R}^3$ is a vector containing small error angles:

$$s\theta_e = \omega \tag{23}$$

$$sJ\omega = -PI(s)\tilde{K}_R\theta_e - PID(s)\omega + \tau_e \tag{24}$$

in which

$$\tilde{K}_R := \begin{bmatrix} \frac{K_{R_2}+K_{R_3}}{2} & 0 & 0 \\ 0 & \frac{K_{R_1}+K_{R_3}}{2} & 0 \\ 0 & 0 & \frac{K_{R_1}+K_{R_2}}{2} \end{bmatrix}, \tag{25}$$

The non-trivial step in deriving the linearized closed-loop system is the computation of equivalent gain matrix $\tilde{K}_R$ (25), which can be obtained by linearizing function $e_R(R_e)$ defined in (20) for $R_e \approx I_3 + \hat{\theta}_e$ as follows. Then, the approximation of (20) reads

$$e_R(R_e) \approx \frac{K_R\hat{\theta}_e - \hat{\theta}_e^T K_R}{2} = \frac{K_R\hat{\theta}_e + \hat{\theta}_e K_R}{2}. \tag{26}$$

Finally, by exploiting the property of the hat map $A^T\hat{x} + \hat{x}A = ((\mathrm{tr}(A)I_3 - A)\,x)^\wedge$ for any $A \in \mathbb{R}^{3\times3}$, $x \in \mathbb{R}^3$, equation (26) can be compactly written as*:

$$e_R(R_e) \approx \tilde{K}_R\theta_e. \tag{27}$$

Since the integrator state is $e_{I_i}(t) := \int_0^t (\tilde{K}_{R_i}\theta_{e_i} + \omega_i)\mathrm{d}t$, by defining the change of coordinates $\bar{e}_{I_i} := e_{I_i} - \frac{1}{K_{I_i}}\tau_{e_i}$, system (23)-(24) can be then written as three decoupled set of equations of the following form:

$$\dot{\theta}_{e_i} = \omega_i \tag{28}$$

$$(J_i + K_{D_i})\dot{\omega}_i = -K_{p_i}(\tilde{K}_{R_i}\theta_{e_i} + \omega_i) - K_{I_i}\bar{e}_{I_i} \tag{29}$$

$$\dot{\bar{e}}_I = (\tilde{K}_{R_i}\theta_{e_i} + \omega_i). \tag{30}$$

By direct substitution, the closed-loop solutions of the linearized system (28)-(24) evolve according to $(J_i + K_{D_i})\dddot{\theta}_{e_i} + K_{P_i}\ddot{\theta}_{e_i} + (K_{I_i} + \tilde{K}_{R_i}K_{P_i})\dot{\theta}_{e_i} + \tilde{K}_{R_i}K_{I_i}\theta_{e_i} = 0$, whose characteristic polynomial is:

$$(J_i + K_{D_i})s^3 + K_{P_i}s^2 + (K_{I_i} + \tilde{K}_{R_i}K_{P_i})s + \tilde{K}_{R_i}K_{I_i}. \tag{31}$$

Hence, to conclude local asymptotic stability we invoke [24, Theorem 4.7] according to which the gains of the diagonal matrices must be selected in order to make the polynomial in (31) Hurwitz for $i = 1, 2, 3$. Since for a given set of polynomial coefficients we can always find a corresponding set of gains, by Routh-Hurwitz stability criterion for

*Matrix $\tilde{K}_R$ has a non-trivial relationship with matrix K_R, which is the one actually used in the control law. For instance, when considering small attitude errors, increasing the term K_{R_3} does not correspond to a larger gain relative to yaw but rather to an increase of the roll and pitch gains.

third-order polynomials there exist gains such that all the roots of (31) have strictly negative real part, meaning that the equilibrium $(R, \omega, \bar{e}_I) = (R_d, 0, 0)$ of (28)-(30), or equivalently, $(R, \omega, e_I) = (R_d, 0, K_i^{-1}\tau_e)$ for the original system (23)-(24), is locally asymptotically stable, which concludes the proof. □

Remark 4 *The suggested controller has been shown to guarantee local asymptotic tracking of any constant reference. As it does not rely on any parametrization of* SO(3)*, the control law is well defined for any attitude configuration and one may be interested in understanding how large is the region of attraction. Since* (18) *is smooth and time-invariant, the best results that one can look for is almost global stability*[†] *While we cannot prove formally this result, it is reasonable to believe that the proposed control law will work well in practice as the following reasoning shows. Let us consider the function* $\Psi(R_e) := \frac{1}{2}tr(K_R(I - R_e))$*, which is a continuously differentiable, positive definite function on* SO(3)*. Its time derivative along the flows of the closed-loop system reads:*

$$\begin{aligned}\dot{\Psi}(R_e) &= \frac{1}{2}tr(K_R\dot{R}_e) = \frac{1}{2}tr(K_R\dot{R}_d^T R + R_d^T\dot{R}))\\ &= \frac{1}{2}tr(K_R R_d^T R\hat{\omega}) = \frac{1}{2}tr(K_R R_e\hat{\omega})\\ &= \frac{1}{2}tr((K_R R_e - R_e^T K_R)\hat{\omega}) = e_R^T(R_e)\omega. \end{aligned} \tag{32}$$

If the control torque (19) *could track the desired value* ω_v *sufficiently fast (instantaneously with respect to the outer loop dynamics), then* $\omega \approx \omega_v$ *and equation* (32) *would read:*

$$\dot{\Psi}(R_e) \approx e_R^T(R_e)\omega_v = -\|e_R(R_e)\|^2, \tag{33}$$

which is negative $\forall R_e \in$ SO(3) *except at* $R_e = I_3$ *and three other points in* SO(3)*, namely,* $R_e = R_{e_i}(k\pi_i)$ *where* $k \in \mathbb{Z}$ *and* $\forall i \in \{1, 2, 3\}$*[22]. Essentially, when assuming that the inner closed-loop is sufficiently fast in tracking the desired angular velocity* ω_v*, the system behaves approximately like the kinematic error model* $\dot{R}_e = -R_e\hat{e}_R(R_e)$*, for which it is well known [22, 23] that the equilibrium point* $R_e = I_3$ *is almost globally asymptotically stable. Note that the separation assumption between the inner and outer loop is not restrictive in practice since the motor dynamics is sufficiently fast for small scale UAVs with respect to desirable closed-loop performance that is typically required for the attitude dynamics of such platforms.*

Remark 5 *As shown in the stability analysis presented above, the control law* (18)-(19) *solves Problem 1 only for the*

[†] An equilibrium point is said to be almost-globally asymptotically stable if the set of initial conditions converging to undesired equilibria is of zero measure, in the sense of Lebesgue, with respect to the manifold SO(3) [22].

case of constant reference ($\omega_d(t) = 0$). This issue can be resolved by considering

$$\omega_v := -e_R(R_e) + R_e^T \omega_d, \tag{34}$$

which is (18) *with an additional term including the desired angular velocity ω_d. Note, however, that this signal is not always available, especially when the UAV is manually piloted, and that it complicates the overall control architecture when the position stabilization loop is included (see the discussion in Remark 7 for additional details). Nonetheless, it will be shown in the linearized closed-loop analysis presented below that even without the feedforward term, the control law* (18)-(19) *can be tuned to achieve a quite large bandwidth in near hovering conditions. We will verify experimentally that satisfactory performance can be obtained for large attitude changes with the same tuning.*

Remark 6 *Hybrid synergistic designs [25–27] or quaternion-based controllers [28] could be employed as well for the outer loop control (equation* (3)*) to avoid having undesired equilibria in the closed-loop system.*

B. Linearized closed-loop system for control law tuning

Following up on the reasoning in Remark 2, the angular velocity dynamics of small scale UAVs is well represented by three independent equations. To achieve satisfactory closed-loop performance without annoying manual tuning procedures, it is advisable to have a sufficiently accurate knowledge of the transfer functions $G_{\omega_i}(s)$ (at least in the frequency range of interest) so that one can apply systematic tuning methods on the linearized closed-loop system [29]. Given $G_{\omega_i}(s)$, the inner closed-loop function (from virtual angular rate ω_{v_i} to ω_i) is

$$\frac{\omega_i}{\omega_{v_i}} = \frac{G_{\omega_i}(s)PI(s)}{1 + G_{\omega_i}(s)PID(s)} =: T_{\omega_i}(s) \tag{35}$$

while the outer closed-loop function, also referred to as the complementary sensitivity function, is

$$\frac{\theta_i}{\theta_{d_i}} = \frac{\frac{1}{s}\tilde{K}_{R_i} T_{\omega_i}(s)}{1 + \frac{1}{s}\tilde{K}_{R_i} T_{\omega_i}(s)} =: T_{\theta_i}(s). \tag{36}$$

From (36), the sensitivity function, which can be interpreted as the closed-loop transfer function from the disturbance d_i to θ_i (see again Figure 3), reads:

$$S_{\theta_i}(s) = 1 - T_{\theta_i}(s). \tag{37}$$

At this point, one can either tune the inner loop gains first and then select the gain $\tilde{K}_{R_i}$, according to the cascade structure in (23)-(35), or directly work with the sensitivity and complementary sensitivity functions in equations (37) and (36). In this work the latter approach has been followed and structured H_∞ synthesis will be used to tune the gains, see Section VII. It is worth mentioning that the flexibility offered by the PID control law proposed in (19) allows shaping

the complementary sensitivity function with sufficient ease for the typical transfer functions $G_{\omega_i}(s)$ that one can expect from small scale multirotor UAVs. The block diagram of the linearized control architecture is depicted in Figure 3.

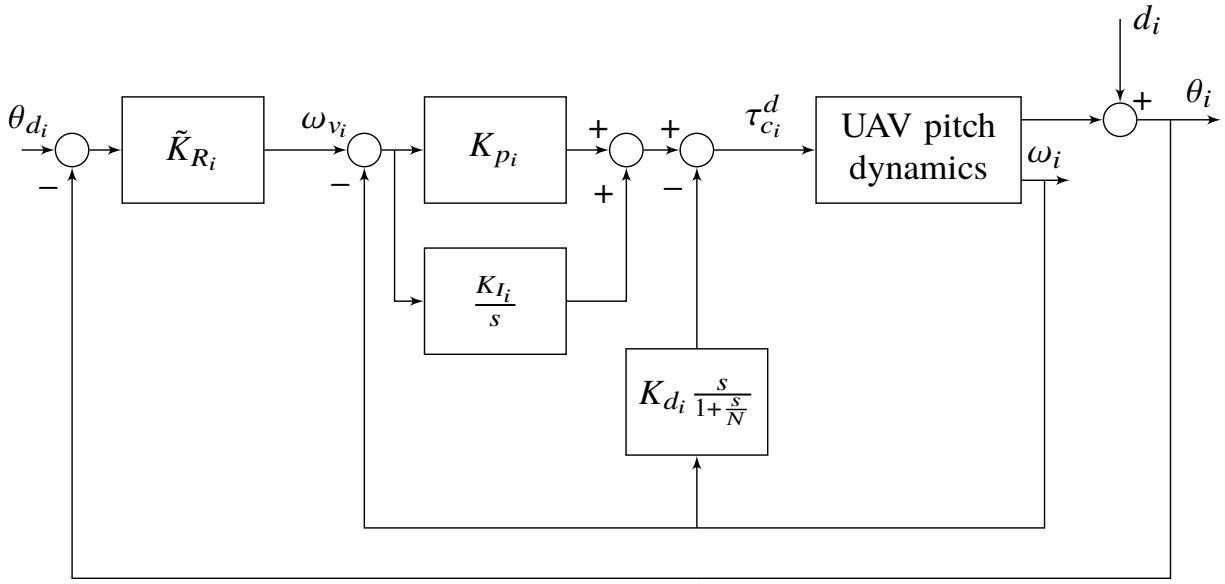

Fig. 3 Linearized cascade architecture (single axis).

VI. Geometric PI-like architecture

We consider a geometric PI-like control law that is inspired by [18] for comparison purposes and to validate the tuning procedure (to be presented in the next section) on a different control architecture. It is derived by means of Lyapunov arguments by referring to the dynamics (2), (11) in which the disturbance torque is assumed to be unknown but constant. Several versions of this control law have been proposed and experimentally validated [18, 30].

A. Control law

The following PI-like geometric control law is considered:

$$\tau_c := -e_R(R_e) - K_\omega e_\omega - K_I e_I + J R_e^T \dot{\omega}_d + (R_e^T \omega_d) \times J R_e^T \omega_d \tag{38}$$

$$\dot{e}_I := e_R(R_e) + K_{\omega_I} e_\omega \tag{39}$$

where $e_R(R_e)$ is defined as for (18), $e_\omega := \omega - R_e^T \omega_d$ and $K_\omega, K_I, K_{\omega_I} \in \mathbb{R}^{3\times3}$ are diagonal positive definite matrices. With respect to [18] we introduced the gain matrix K_{ω_I} in front of the angular velocity error in the dynamics of the integral term (39) in order to have more freedom in weighting the different terms. By exploiting Lyapunov arguments [18], it is possible to show that there exist diagonal gain matrices $K_R, K_\omega, K_I, K_{\omega_I}$ such that the closed-loop

equilibrium $(R_e, e_\omega, e_I) = (I_3, 0, K_I^{-1}\tau_e)$ of the error dynamics

$$\dot{R}_e = R_e \hat{e}_\omega \tag{40}$$

$$J\dot{e}_\omega = -K_\omega e_\omega - e_R(R_e) + (Je_\omega + dR_e^T \omega_d) \times e_\omega - K_I e_I + \tau_e, \tag{41}$$

$$\dot{e}_I = e_R(R_e) + K_{\omega_I} e_\omega, \tag{42}$$

where $d := 2J - \mathrm{tr}(J)I_3$, is locally exponentially stable.

B. Linearized closed-loop system for control law tuning

In this section the closed-loop system for the PI-like control law is presented and discussed. Before proceeding, it is worth noting that by (39), the PI-like control law includes the integration of the angular velocity error which, contrary to the linear case, is not just equivalent to increase the proportional gain in (38) unless small attitude errors are considered. Indeed, when $R_e \approx I_3 + \hat{\theta}_e$, $e_\omega \approx 0$, the linearized closed-loop error system is:

$$s\theta_e = e_\omega \tag{43}$$

$$sJe_\omega = -K_\omega e_\omega - \tilde{K}_R \theta_e - K_I e_I + \tau_e \tag{44}$$

$$se_I = \tilde{K}_R \theta_e + K_{\omega_I} e_\omega \tag{45}$$

in which $\tilde{K}_R$ is defined as in (25) and the gyroscopic terms in the dynamic equation (3) have been neglected, consistently with the analysis of the cascade design. Equations (44) and (45) can be rearranged as

$$sJe_\omega = -K_\omega e_\omega - \tilde{K}_R \theta_e - K_I \frac{1}{s}\left(\tilde{K}_R \theta_e + K_{\omega_I} e_\omega\right) + \tau_e. \tag{46}$$

To determine the local stability behavior of the closed-loop system, one can refer to the corresponding characteristic polynomial related to the i-th axis, namely,

$$s^3 J_i + s^2 K_{\omega_i} + s(\tilde{K}_{R_i} + K_{I_i} K_{\omega_{I_i}}) + K_{I_i} \tilde{K}_{R_i} \tag{47}$$

in which τ_e can be neglected since it is an exogenous signal and does not contribute to the stability of the closed-loop system. To ensure local asymptotic stability, the gains of the diagonal matrices must be selected in order to make the polynomial in (47) Hurwitz for $i = 1, 2, 3$. A different path is to exploit the knowledge of identified models $G_{\omega_i}(s)$, as done for the cascade architecture, and tune the gains in order to achieve satisfactory performance when referring to the

sensitivity function from d_i to θ_i, given by:

$$S_{\theta_i}(s) := \frac{1}{1 + \frac{1}{s} T_{\omega_i}(s) C_{\theta_i}(s)} = \frac{\theta_i}{d_i} \tag{48}$$

where

$$\begin{aligned} T_{\omega_i}(s) &:= \frac{G_{\omega_i}(s)}{1 + G_{\omega_i}(s) C_{\omega_i}(s)} \\ C_{\omega_i}(s) &:= K_{\omega_i} + \frac{1}{s} K_{I_i} K_{\omega I_i} \\ C_{\theta_i}(s) &:= \tilde{K}_{R_i} \left(1 + \frac{1}{s} K_{I_i}\right). \end{aligned}$$

The block diagram of the linearized geometric PI-like architecture for a single axis is depicted in Figure 4.

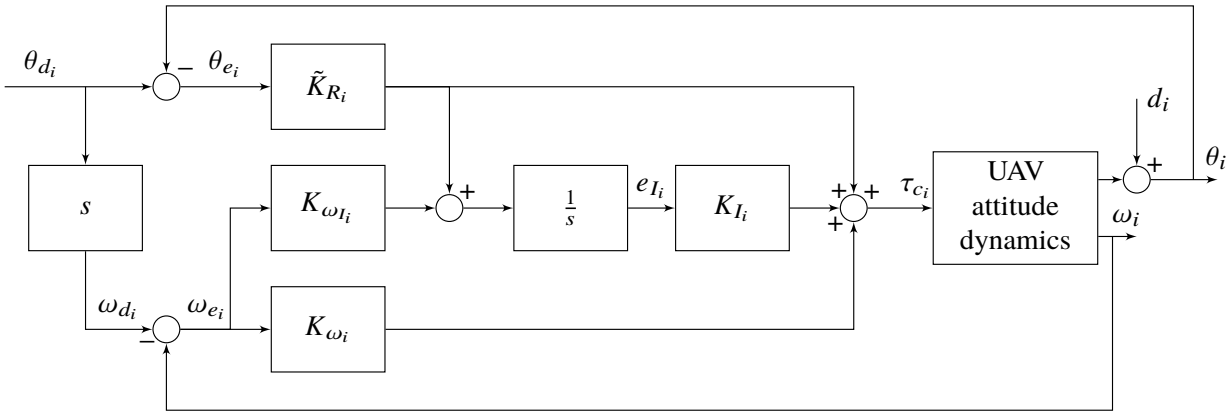

Fig. 4 Linearized geometric PI-like control architecture (single axis).

Remark 7 *The control torque in equation* (38) *requires both the desired angular velocity and acceleration to be implemented. In the position tracking scenario depicted in Section III this means that the position controller should provide not only a desired attitude R_d but ω_d and $\dot{\omega}_d$ as well. However, computing analytically ω_d and $\dot{\omega}_d$ requires continuously differentiable position trajectories up to the fourth order [6]. This not only makes the controller structure more complex but has potentially negative effects when only a roughly estimated inertia matrix is available. Furthermore, in case the vehicle is manually piloted, the pilot sends commands in terms of desired angles to the on-board controller and the corresponding angular velocity and acceleration must be somehow computed on-line, unlike the scenario in which both the attitude and its derivatives are provided, for instance, by a ground control computer. One option is to set $\omega_d, \dot{\omega}_d = 0$, namely, set-point tracking. This, however, represents a limitation in achievable performance in terms of attitude response to reference, resulting in a sluggish response. In practice this work adopts the approach of [19, 31], where* (38) *is implemented with good approximation as $\tau_c^d \approx -e_R(R_e) - K_\omega e_\omega - K_I e_I$ which does not depend upon the UAV inertia and does not require the reference $\dot{\omega}_d$. Note that when the approximated torque is linearized it yields*

exactly the closed-loop equation (44)*. Such control torque still requires the desired angular velocity* ω_d *to compute the error* e_ω*, therefore a continuously differentiable signal with its derivative must be provided to the controller. This issue is addressed in detail in the next section.*

C. Reference signal related issues

In Remark 7 it has been pointed out that the geometric PI-like architecture requires the desired attitude R_d and at least the corresponding angular velocity ω_d in the approximated case which neglects the feed-forward contribution in (38). Following up on the comments presented above, a smooth trajectory generator is therefore needed to provide the controllers with that information and to carry out a fair comparison between the two architectures. The trajectory generator has been implemented on-board in the form of a filter so that the existing software architecture should not be modified: the idea is to pass the pilot/computer command through a suitable filter, so that a continuously differentiable signal and its derivative (at least) can be provided to the controllers. Since both controllers are geometric, the filter should be developed directly on SO(3). To this end, we propose the geometric counterpart of the Euclidean second-order filter $\ddot{\theta}_d^f = -\omega_n^2(\theta_d^f - \theta_d) - 2\xi\omega_n\dot{\theta}_d^f$ $(\omega_n, \xi \in \mathbb{R}_{>0})$ which, in transfer function form, can be written as:

$$\begin{bmatrix} \theta_d^f \\ \omega_d^f \end{bmatrix} = \begin{bmatrix} F_\theta(s) \\ sF_\theta(s) \end{bmatrix} \theta_d = F(s)\theta_d, \qquad F_\theta(s) := \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}. \tag{49}$$

Specifically, the following geometric filter has been developed:

$$\dot{R}_d^f = R_d^f \hat{\omega}_d^f \tag{50}$$

$$\dot{\omega}_d^f = -\omega_n^2 e_R(R_e^f) - 2\xi\omega_n\omega_d^f \tag{51}$$

where $R_e^f := R_d^T R_d^f \in \mathrm{SO}(3)$ and $e_R(\cdot)$ is defined as in (19). It can be verified that (49) is the linearized version of (50)-(51) for a small attitude motion $t \mapsto R_d(t) \approx I_3 + \hat{\theta}_d(t)$.

VII. Tuning

The linearized version of the two control architectures presented in Sections V and VI, respectively, were used to carry out tuning of the gains. This allows one to resort to systematic tuning methods for linear systems. In particular, in this work the structured H_∞ approach is considered. The tuned gains are then plugged in the respective nonlinear control law architectures for validation in the time domain. The approach is shown for only the pitch axis, however, it can be straightforwardly applied to the other axes. In order to ease the notation, the superscript i will be dropped, referring to the pitch axis (*i.e.*, $i = 2$).

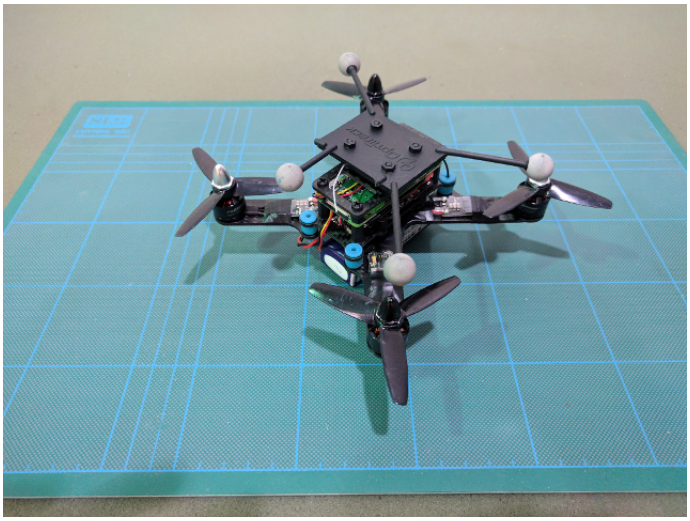

Fig. 5 Quadrotor used for the tests.

A. Hardware set-up

In this work, the quadrotor depicted in Figure 5 was considered; it is a lightweight custom model with a distance of 160mm between opposite rotor axes and an overall take-off weight of about 230g. The relevant parameters are reported in Table 1. The flight control unit is a Pixfalcon board, an open autopilot shield suitable for remotely controlled vehicles such as multirotors and fixed wing aircraft. It is equipped with a 3 axes accelerometer, a 3 axes gyroscope, a magnetometer and a pressure sensor. The firmware running on the Pixfalcon board is the open-source software PX4 Pro Autopilot. The firmware features attitude and position controllers and estimators, and has been customized to allow replacing the baseline attitude controller with user-defined controllers.

Table 1 Main quadrotor parameters.

Variable	Value
Frame Config.	X
Propellers	Gemfan Bullnose 3035 3 blade
Arm length b	80 mm
Take-off weight m	230 g
Motors	QAV1306-3100kV brushless
ESC	ZTW Spider series 18A
Battery	Turnigy nano-tech 950mAh LIPO

B. Model identification

A black-box model of the pitch attitude dynamics of the UAV was identified with the PBSID subspace model identification algorithm using closed-loop experimental data [32] and was used as the basis for control law tuning.

The identified model, from the (adimensionalized) pitch moment input M to the pitch rate output q, is of order five and contains a time delay. Figure 6 shows the frequency response of the identified model against the estimate of the non-parametric frequency response function (computed accounting for the bias introduced by feedback in the closed-loop experiment). The coherence function indicates that the experimental data is valid in the frequency range from 10 rad/s to 100 rad/s, which is consistent with the expected attitude control bandwidth. Both magnitude and phase of the identified model feature excellent fit to the nonparametric frequency response in this frequency range. The model was validated against flight data collected in a manually piloted experiment: Figure 7 shows the measured response (blue line) to the pitch attitude reference signal (red line), against the simulated closed-loop attitude angle response (black line) obtained with the identified model, showing a close match to the measured data.

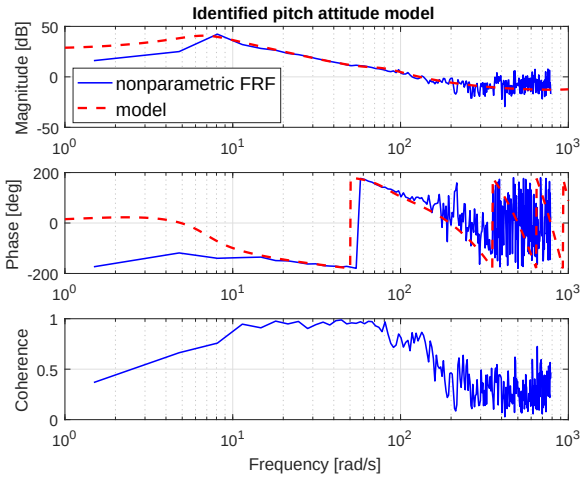

Fig. 6 Frequency response of the identified pitch attitude model

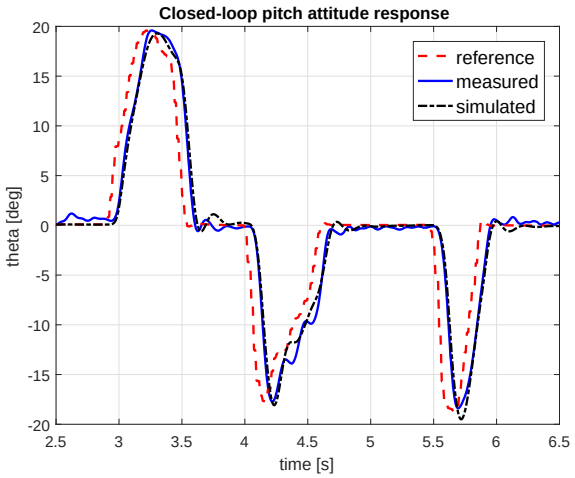

Fig. 7 Validation of the identified model.

C. Requirements definition

The control requirements for mixed-sensitivity structured H_∞ synthesis are stated in the form of weighting functions in the frequency domain. In particular, weighting functions are rational, stable, proper transfer functions. Two different requirements were taken into account:

- performance: the requirement on performance is defined as a weighting function on the attitude angle sensitivity, *i.e.*, the transfer function from the disturbance on the pitch angle d to θ (compare with equations (37) and (48) for the cascade and the geometric PI architectures, respectively);
- control moderation: the requirement on control moderation is defined as a weighting function on the control sensitivity, *i.e.*, the transfer function from d to M.

It was decided to characterize performance in terms of disturbance rejection [14, 33] (*i.e.*, breaking the loop in the output), rather than in terms of the response to the reference input (*i.e.*, breaking the loop in correspondence of the

Table 2 Sensitivity weighting function parameters.

	ω_{DRB} rad/s	K_{HF} dB	K_{LF} dB	k_S	p_S	z_S
H_{BW}	9	8	-40	0.398	0.122	30.64
L_{BW}	2	3	-40	0.708	0.02445	3.454

tracking error signal): indeed, the latter inherently depends both on the feedback and feed-forward properties of the system, while the former only depends on the feedback regulator. This approach allows comparing the performance of the two considered control architectures in terms only of the feedback properties (see also the discussion in Remark 7).

The weighting function on the sensitivity is defined as:

$$W_S(s) := k_S \frac{s + z_S}{s + p_S} \tag{52}$$

where the inverse of the frequency response magnitude $\left|W_S(j\omega)^{-1}\right|$ represents an upper bound on the sensitivity frequency response magnitude $|S(j\omega)|$; the parameters of the weighting function are chosen so as to enforce a low-frequency constraint $\left|W_S(j0)^{-1}\right| = K_{LF}$, a high-frequency constraint $\left|W_S(j\infty)^{-1}\right| = K_{HF}$, and a bandwidth constraint $\left|W_S(j\omega_{DRB})^{-1}\right| = -3\,\text{dB}$. In particular, ω_{DRB} represents the desired sensitivity bandwidth, also referred to in the ADS-33 US army requirement specification standard [34] as the disturbance rejection bandwidth (DRB), while K_{HF} represents a constraint on the peak of the sensitivity magnitude, also referred to as the disturbance rejection peak (DRP) [33, 34].

Two different levels of performance were defined:

- a *high-bandwidth* requirement (H_{BW}), representing aggressive performance requirements, with a large bandwidth and allowing for a large sensitivity peak;
- a *low-bandwidth* requirement (L_{BW}), featuring a more stringent constraint on the sensitivity peak, on the other hand trading off a lower bandwidth.

The parameters of the corresponding weighting functions are reported in Table 2.

The control sensitivity weigthing function was chosen as follows:

$$W_R(s) := k_R \frac{s + z_R}{s + p_R} \tag{53}$$

with $k_R = 0.2$, $p_R = 23.26$ and $z_R = 10^{-4}$, in order to limit high frequency control action beyond the actuators bandwidth.

D. Control law synthesis problem statement

Consider the vector of tunable controller parameters, which for the cascade architecture is $\rho_C := \left[K_{P_i}, K_{I_i}, K_{D_i}, \tilde{K}_{R_i}\right]^T$ and for the geometric PI architecture is $\rho_G := \left[\tilde{K}_{R_i}, K_{\omega_i}, K_{I_i}, K_{\omega_{I_i}}\right]^T$, where $i = 1, 2, 3$ indicates respectively the roll, pitch or yaw axis, and the control law tuning is carried out one axis at a time. Let $S(s, \rho)$ be the sensitivity function and $R(s, \rho)$ the control sensitivity function, where the dependence on ρ is made explicit. Let $J_S(\rho)$ be the cost function related to the performance requirement, and let $J_R(\rho)$ be the cost function related to the control moderation requirement:

$$J_S(\rho) := \|W_S(s)S(s, \rho)\|_\infty \tag{54}$$

$$J_R(\rho) := \|W_R(s)R(s, \rho)\|_\infty . \tag{55}$$

The synthesis problem can be stated as an optimization problem:

$$\rho^* = \arg\min_\rho \; J_S(\rho) \tag{56}$$

$$\text{subject to} \tag{57}$$

$$J_R(\rho) \leq 1 \tag{58}$$

with ρ^* being the optimal value of the controller gain vector.

E. Numerical results

Three different control law gain sets were obtained for the cascade architecture:

- a *high-bandwidth* cascade control law, denoted as C_H, which is subject to the H_{BW} performance requirement defined in Section VII.C;
- a *low-bandwidth* cascade control law, denoted as C_L, subject to the L_{BW} performance requirement defined in Section VII.C;
- a *low-bandwidth* cascade control law with no derivative action in the inner loop, denoted as C_L^{D0}, which is subject to the same L_{BW} performance requirement as the C_L controller, but with the additional constraint $K_D = 0$.

Numerical optimization was carried out by means of the MATLAB `systune` routine. The gain values for the three cascade control laws are shown in Table 3, along with the achieved sensitivity bandwidth ω_{DRB} and complementary sensitivity ω_T.

Finally, gains for the geometric PI architecture were tuned according to the L_{BW} performance requirement and are stated in Table 4; this control law is denoted as G_L. Unlike the cascade architecture, it was not possible to obtain a solution achieving the H_{BW} requirements with the geometric PI control architecture, likely due to the absence of a derivative action on the angular rate.

Table 3 Cascade control architecture gain values: pitch axis ($i = 2$).

Gain	C_H	C_L	C_L^{D0}
K_P	0.142	0.187	0.0646
K_I	0.287	0.362	0.107
K_D	0.00263	0.00344	0
$\tilde{K}_R$	12.5	3.30	3.10
ω_{DRB} [rad/s]	10.0	2.98	2.26
ω_T [rad/s]	35.9	3.26	2.47

Table 4 Geometric PI control architecture gain values: pitch axis ($i = 2$).

Gain	G_L
$\tilde{K}_R$	0.264
K_ω	0.0757
K_I	0.3
K_{ω_I}	0.0390
ω_{DRB} [rad/s]	2.83
ω_T [rad/s]	39.65

Figure 8 shows the sensitivity function magnitude of the system closed in loop with controllers C_L, C_L^{D0} and G_L, along with the inverse of the weighting function associated with the low performance requirement; in all the cases, the optimization routine is able to meet the constraint. The effect of the derivative action on angular rate can be appreciated in that the C_L controller achieves a significantly lower sensitivity magnitude peak with respect to the other two controllers (which are not provided with angular rate derivative action), and a larger bandwidth with respect to the required one.

Figure 9 shows the Bode magnitude plot of the complementary sensitivity function for the four controllers. It can be noticed that the C_L and C_L^{D0} controllers achieve a similar complementary sensitivity magnitude shape, while the C_H controller achieves higher bandwidth to reference response, as expected; on the other hand, the complementary sensitivity for the G_L controller features a magnitude shape closer to the high-bandwith C_H controller rather than the cascade low-bandwith controllers C_L and C_L^{D0}, despite having been designed for low-bandwidth requirements. It is thus expected that the G_L controller achieves a fast response, though with some oscillations (due to the presence of a resonance peak). Furthermore, from the phase plot it can be noticed that G_L features the smallest phase lag among the four controllers.

The filter $F(s)$ of equation (49), described in Section VI.C, was appended upstream the geometric PI controller, and the part of filter $F_\theta(s)$ related to attitude angle reference was appended upstream the cascade controller; in this way, a fair comparison between the two control architectures can be carried out, in terms of response to piloted attitude

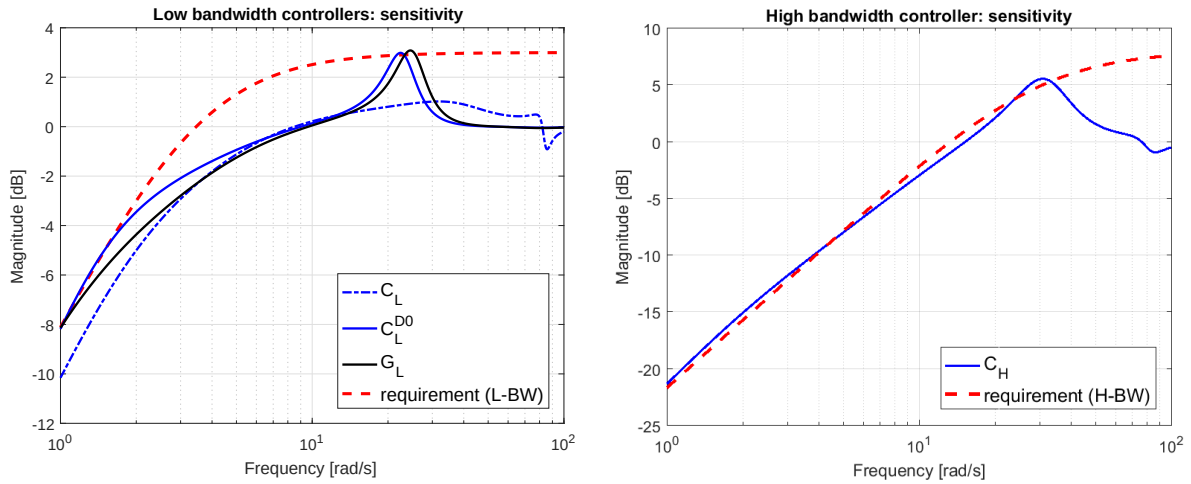

Fig. 8 Sensitivity functions for the different controllers

angle reference. The step responses of the system closed in loop with the four controllers are shown in Figure 10, along with the response of the filter $F_\theta(s)$ to a step reference, which itself acts as a reference signal to be tracked by the four controllers.

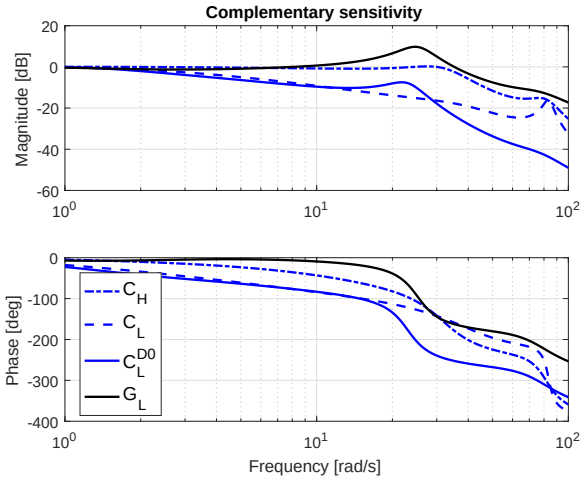

Fig. 9 Complementary sensitivity functions for the different controllers.

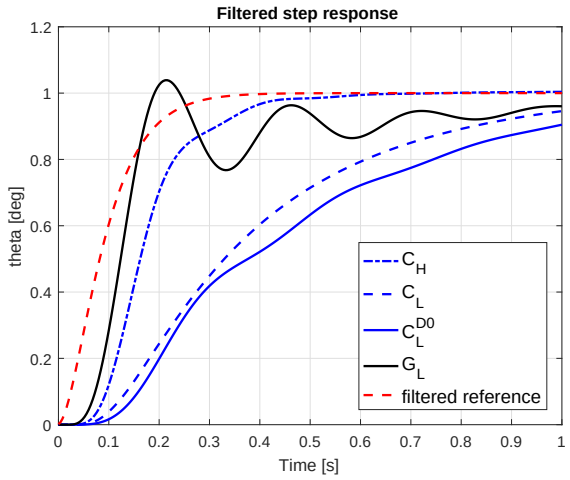

Fig. 10 Response to filtered step angle reference.

VIII. Experimental results

In this section the tuning technique is validated on the quadrotor platform described in Section VII.B and a thorough experimental comparison between the two proposed architectures is shown.

First of all, we verify that the actual closed-loop systems have a behavior comparable with the analysis of the previous section by considering test-bed experiments in which all the degrees of freedom except pitch rotation are constrained. To this end, three experiments were designed to test the closed-loop response to the attitude angle reference:

- a series of doublets at small attitude angle, to test the system behaviour close to the equilibrium condition;
- a series of doublets at large attitude angle, to test the system behaviour far from the equilibrium;
- a sweep signal to excite the closed-loop system within the control bandwidth, to assess tracking performance.

The doublet half-period is $T = 0.5\,\text{s}$, with an amplitude of $A = 10\,\text{deg}$ for the small-amplitude experiment and $A = 30\,\text{deg}$ for the large-amplitude experiment; the sweep signal has an amplitude $A = 10\,\text{deg}$ and excites the system in the frequency range from $1.4\,\text{rad/s}$ to $14\,\text{rad/s}$.

Figure 11 shows a comparison between the attitude angle responses of the system closed in loop with the C_H and C_L controllers to the large-amplitude doublet input. Consistently with the simulation (compare with Figure 10), the C_H controller achieves a faster response than C_L, at the price of slight oscillations in the response; this can be interpreted in the frequency domain as a trade-off between bandwidth and damping ratio.

Figure 12 shows a comparison between the doublet responses obtained with the controllers C_L^{D0} and G_L. These two controllers are comparable from the point of view of bandwidth and do not include any derivative action on the attitude rate; however, the response of G_L is more aggressive and presents many oscillations, which is consistent with the expected behavior of the linearized closed-loop system obtained in Section VII.E. The response of controllers C_L and C_L^{D0} is very similar.

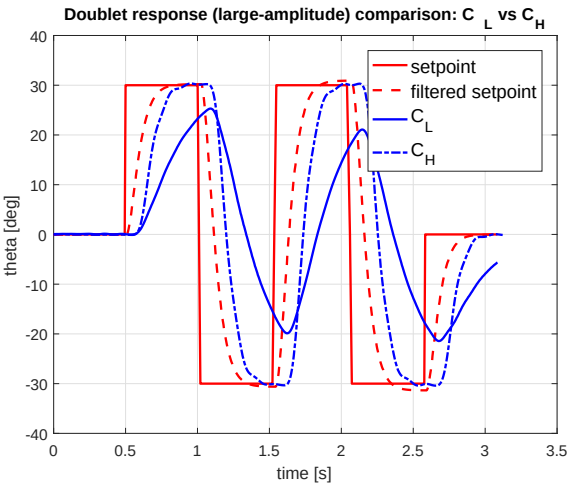

Fig. 11 Response to doublets: controllers C_H and C_L.

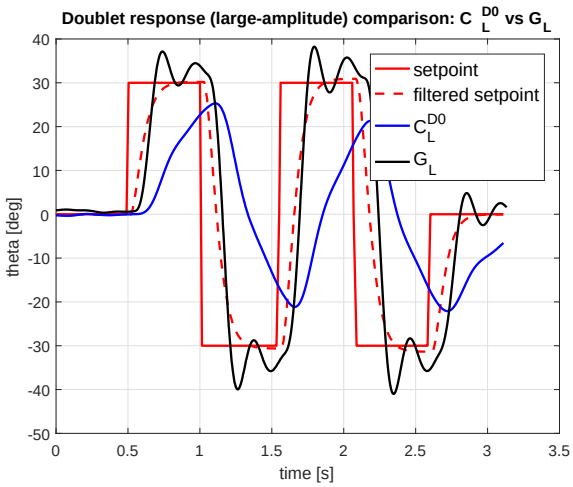

Fig. 12 Response to doublets: controllers C_L^{D0} and G_L.

Figure 13 shows a comparison between the sweep responses obtained with the G_L, C_L^{D0} and C_H controllers, with focus on the initial time instants. Note that the sweep reference is filtered by (49) so that the signal is subject to a roll-off in magnitude as frequency increases. By inspecting Figure 13, the geometric PI-like architecture achieves better tracking than the cascade architectures, both in terms of amplitude of the response and limited phase lag with respect to the reference signal. As expected, the C_L^{D0} controller features a significant lag in the response, even at low frequencies, and a quick decrease in peak magnitude as the frequency increases. On the contrary, the C_H achieves satisfactory

performance and only a minimum lag with respect to the G_L controller. However, by looking at Figure 14, the response of the G_L shows a resonance effect as the sweep frequency increases with time, resulting in an attitude response even larger than the reference signal. This is again expected by looking at the complementary sensitivity function of the G_L controller in Figure 9.

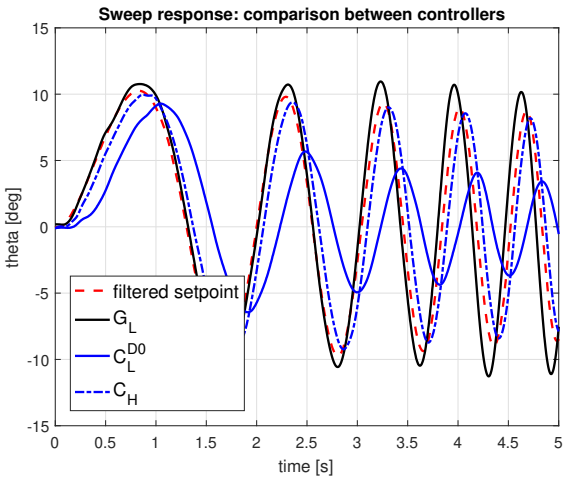

Fig. 13 Sweep response: comparison of controllers G_L, C_L^{D0}, C_H

.

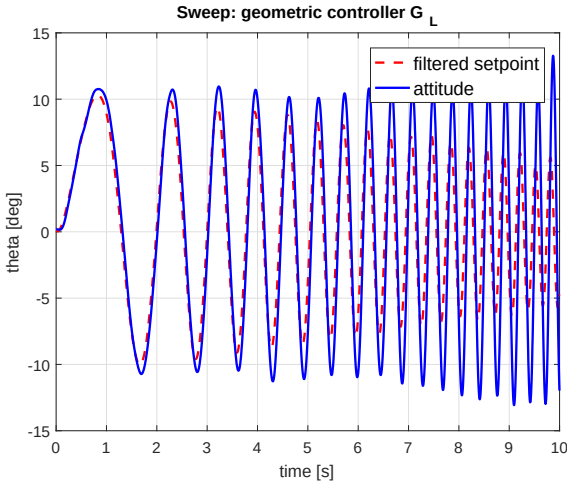

Fig. 14 Sweep response: controller G_L

.

Overall, the experiments confirm the validity of the proposed tuning methodology since the responses obtained with the different controllers are in good agreement with the expected behavior according to the complementary sensitivity functions. Furthermore, the test results, conducted at large amplitude angle, showed an attitude response comparable to the small amplitude experiments (not shown for conciseness); this confirms that the two nonlinear control laws are able to handle large displacements from the equilibrium condition without impairing performance.

In addition to test-bed experiments, flight tests were carried out on the unconstrained quadrotor in order to show that the decoupled controller strategy can provides good results, even for highly aggressive maneuvers. First of all, several flight tests were performed with the UAV manually piloted in attitude control mode. The different controllers were tuned on both the pitch and roll axis. This experiments confirmed the results obtained on the test-bed, with all the controllers guaranteeing stability and acceptable maneuverability from the pilot's point of view. Based on the pilot's opinion, the best performance in terms of speed of response is obtained with the C_H controller, while the G_L controller features some oscillations, mitigated by reducing the bandwidth of the set-point filter. This, however, is a trade-off between ease of flight and speed of response.

Finally, a series of doublets at 45 deg was given as a reference for both the roll and pitch angle. The experiment was carried out with the C_H controller, which was the best candidate for such a challenging test. To cope with a slight asymmetry in the mass distribution, the robust tuning procedure outlined in Section VII was applied to the

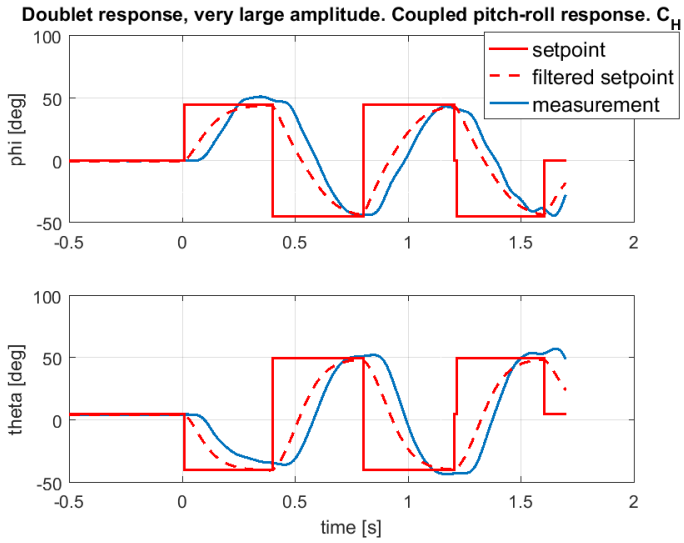

Fig. 15 Combined in-flight roll and pitch doublet responses: C_H controller.

roll axis as well (not shown for the sake of conciseness). Figure 15 confirms that the controller performance is only slightly deteriorated with respect to the single axis experiments: it is worth remarking that a larger maximum angle is commanded with respect to the single axis experiments and that flight tests in attitude control are characterized by translational motion, which can induce significant aerodynamic disturbances.

IX. Concluding remarks

In this paper we presented a systematic methodology, based on structured H_∞, to tune the gains of nonlinear control laws for attitude control in multirotor UAVs. Moreover, a cascade P/PID-like architecture was developed in the framework of geometric control theory and was compared with a nonlinear PI-like controller borrowed from the literature. The tuning methodology was validated on both control architectures: the proposed cascade controller was easily tuned to achieve a desired level of performance and a major advantage over the PI controller, in terms of architecture complexity, is that it requires only a desired attitude signal to be implemented on-board, which has two practical implications. First, when included in a hierarchical strategy for position tracking, it does not require the computation of a desired angular velocity and acceleration, which depends upon the snap of the desired position trajectory. Moreover, when manually piloted, the PI-like controller cannot exploit information about the desired angular velocity and acceleration, which significantly reduces the expected performance. A nonlinear trajectory generator filter, which could be easily implemented on-board, has been developed to mitigate such deficiency and to allow for a fair experimental comparison. In this case, as expected by looking at the complementary sensitivity function obtained with the PI-like controller, tests performed with a quadrotor UAV have shown the superior tracking performance over the cascade architecture when the frequency content of the reference signal is not too high. However, the response

is aggressive and features some oscillations which makes it less convenient to be used in manual flight unless the bandwidth of the set-point filter is reduced, which is clearly a trade-off between maneuverability and performance.

References

[1] Panizza, P., Invernizzi, D., Riccardi, F., Formentin, S., and Lovera, M., "Data-driven attitude control law design for a variable-pitch quadrotor," *American Control Conference*, 2016, pp. 4434–4439. doi:https://doi.org/10.1109/ACC.2016.7525620.

[2] Invernizzi, D., Panizza, P., Riccardi, F., Formentin, S., and Lovera, M., "Data-driven attitude control law of a variable-pitch quadrotor: a comparison study," *IFAC-PapersOnLine*, Vol. 49, No. 17, 2016, pp. 236 – 241. doi:https://doi.org/10.1016/j.ifacol.2016.09.041, *20th IFAC Symposium on Automatic Control in Aerospace*.

[3] Hamel, T., Mahony, R., Lozano, R., and Ostrowski, J., "Dynamic modelling and configuration stabilization for an X4-Flyer." *IFAC Proceedings Volumes*, Vol. 35, No. 1, 2002, pp. 217 – 222. doi:https://doi.org/10.3182/20020721-6-ES-1901.00848, 15th IFAC World Congress.

[4] Tayebi, A., and McGilvray, S., "Attitude stabilization of a four-rotor aerial robot," *43rd IEEE Conference on Decision and Control (CDC)*, Vol. 2, 2004, pp. 1216–1221 Vol.2. doi:https://doi.org/10.1109/CDC.2004.1430207.

[5] Lee, T., Leok, M., and McClamroch, H., "Geometric tracking control of a quadrotor UAV on SE(3)," *IEEE Conference on Decision and Control, Atlanta, USA*, 2010. doi:https://doi.org/10.1109/CDC.2010.5717652.

[6] Invernizzi, D., Lovera, M., and Zaccarian, L., "Geometric trajectory tracking with attitude planner for vectored-thrust VTOL UAVs," *2018 Annual American Control Conference (ACC)*, IEEE, 2018, pp. 3609–3614. doi:https://doi.org/10.23919/ACC.2018.8431708.

[7] Naldi, R., Furci, M., Sanfelice, R. G., and Marconi, L., "Robust global trajectory tracking for underactuated VTOL aerial vehicles using inner-outer loop control paradigms," *IEEE Transactions on Automatic Control*, Vol. 62, No. 1, 2017, pp. 97–112. doi:https://doi.org/10.1109/TAC.2016.2557967.

[8] Fernando, T., Chandiramani, J., Lee, T., and Gutierrez, H., "Robust adaptive geometric tracking controls on SO (3) with an application to the attitude dynamics of a quadrotor UAV," *2011 50th IEEE Conference on Decision and Control and European Control Conference*, IEEE, 2011, pp. 7380–7385. doi:https://doi.org/10.1109/CDC.2011.6161306.

[9] Bergamasco, M. and Lovera, M., "Identification of linear models for the dynamics of a hovering quadrotor," *IEEE Transactions on Control Systems Technology*, Vol. 22, No. 5, 2014, pp. 1696–1707. doi:https://doi.org/10.1109/TCST.2014.2299555.

[10] Bressan, G., Russo, A., Invernizzi, D., Giurato, M., Panza, S., and Lovera, M., "Adaptive augmentation of the attitude control system for a multirotor unmanned aerial vehicle," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 2018. doi:https://doi.org/10.1177/0954410018820193.

[11] Jafarnejadsani, H., Sun, D., Lee, H., and Hovakimyan, N., "Optimized L1 adaptive controller for trajectory tracking of an indoor quadrotor," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 6, 2017, pp. 1415–1427. doi: https://doi.org/10.2514/1.G000566.

[12] Kun, D. W., and Hwang, I., "Linear matrix inequality-based nonlinear adaptive robust control of quadrotor," *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 5, 2015, pp. 996–1008. doi:https://doi.org/10.2514/1.G001439.

[13] Apkarian, P., and Noll, D., "Nonsmooth H_∞ synthesis," *IEEE Transactions on Automatic Control*, Vol. 51, No. 1, 2006, pp. 71–86. doi:https://doi.org/10.1109/TAC.2005.860290.

[14] Panza, S., "Structured flight control law design for helicopters and tiltrotors," Ph.D. thesis, Politecnico di Milano, Dipartimento di Scienze e Tecnologie Aerospaziali, 2018.

[15] Navarro-Tapia, D., Simplício, P., Iannelli, A., and Marcos, A., "Robust flare control design using structured-H_∞ synthesis: a civilian aircraft landing challenge," *20th IFAC World Congress, Toulouse, France*, 2017. doi:https://doi.org/10.1016/j.ifacol.2017.08.769.

[16] Lesprier, J., Biannic, J.-M., and Roos, C., "Nonlinear structured H_∞ controllers for parameter-dependent uncertain systems with application to aircraft landing," *IEEE Conference on Control Applications, Antibes, France*, 2014. doi:https://doi.org/10.1109/CCA.2014.6981384.

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

[18] Goodarzi, F., Lee, D., and Lee, T., "Geometric nonlinear PID control of a quadrotor UAV on SE(3)," *Proc. European Control Conf. (ECC)*, 2013, pp. 3845–3850. doi:https://doi.org/10.23919/ECC.2013.6669644.

[19] R. Mahony, V. Kumar and P. Corke, "Multirotor aerial vehicles: modeling, estimation and control of quadrotor," *IEEE Robotics & Automation Magazine*, Vol. 19, No. 3, 2012, pp. 20–32. doi:https://doi.org/10.1109/MRA.2012.2206474.

[20] Mellinger, D., Michael, N., and Kumar, V., "Trajectory generation and control for precise aggressive maneuvers with quadrotors," *Experimental robotics*, Springer, 2014, pp. 361–373. doi:https://doi.org/10.1007/978-3-642-28572-1_25.

[21] Falanga, D., Mueggler, E., Faessler, M., and Scaramuzza, D., "Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision," *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 5774–5781. doi:https://doi.org/10.1109/ICRA.2017.7989679.

[22] Koditschek, D. E., "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, 1989, pp. 131–157.

[23] Chaturvedi, N., Sanyal, A., and McClamroch, N., "Rigid-body attitude control," *IEEE Control Systems*, Vol. 31, 2011, pp. 30–51. doi:https://doi.org/10.1109/MCS.2011.940459.

[24] Khalil, H. K., *Nonlinear systems, 3rd Edition*, Prentice-Hall, 2002.

[25] Mayhew, C. G., Sanfelice, R. G., and Teel, A. R., "Further results on synergistic Lyapunov functions and hybrid feedback design through backstepping," *Proc. 50th IEEE Conf. Decision and Control and European Control Conf*, 2011, pp. 7428–7433. doi:https://doi.org/10.1109/CDC.2011.6161283.

[26] Mayhew, C. G., and Teel, A. R., "Synergistic Hybrid Feedback for Global Rigid-Body Attitude Tracking on SO(3)," *IEEE Transactions on Automatic Control*, Vol. 58, No. 11, 2013, pp. 2730–2742. doi:https://doi.org/10.1109/TAC.2013.2266852.

[27] Berkane, S., Abdessameud, A., and Tayebi, A., "Hybrid global exponential stabilization on $SO(3)$," *Automatica*, Vol. 81, No. Supplement C, 2017, pp. 279 – 285. doi:https://doi.org/10.1016/j.automatica.2017.04.001.

[28] Mayhew, C. G., Sanfelice, R. G., and Teel, A. R., "Quaternion-based hybrid control for robust global attitude tracking," *IEEE Transactions on Automatic Control*, Vol. 56, No. 11, 2011, pp. 2555–2566. doi:https://doi.org/10.1109/TAC.2011.2108490.

[29] Berkane, S., Abdessameud, A., and Tayebi, A., "Hybrid Output Feedback for Attitude Tracking on $\mathbb{SO}(3)$," *IEEE Transactions on Automatic Control*, Vol. 63, No. 11, 2018, pp. 3956–3963. doi:10.1109/TAC.2018.2808445.

[30] Invernizzi, D., Giurato, M., Gattazzo, P., and Lovera, M., "Full pose tracking for a tilt-arm quadrotor UAV," *Proc. IEEE Conf. Control Technology and Applications (CCTA)*, 2018, pp. 159–164. doi:https://doi.org/10.1109/CCTA.2018.8511566.

[31] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotor helicopters," *IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China*, 2011. doi:https://doi.org/10.1109/ICRA.2011.5980409.

[32] van der Veen, G., van Wingerden, J.-W., Bergamasco, M., Lovera, M., and Verhaegen, M., "Closed-loop subspace identification methods: an overview," *IET Control Theory and Applications*, Vol. 7, No. 10, 2013, pp. 1339–1358. doi:http://dx.doi.org/10.1049/iet-cta.2012.0653.

[33] Berger, T., Ivler, C., Berrios, M., Tischler, M., and Miller, D., "Disturbance rejection handling qualities criteria for rotorcraft," *72nd Annual Forum of the American Helicopter Society, West Palm Beach, USA*, 2016.

[34] "ADS-33E-PRF, Aeronautical Design Standard, performance specification. handling qualities requirements for military rotorcraft," , March 2000.