



Direct Acceleration Feedback Control of Quadrotor Aerial Vehicles

Mahmoud Hamandi, Marco Tognon, Antonio Franchi

► To cite this version:

Mahmoud Hamandi, Marco Tognon, Antonio Franchi. Direct Acceleration Feedback Control of Quadrotor Aerial Vehicles. IEEE International Conference on Robotics and Automation (ICRA 2020), May 2020, Paris, France. 10.1109/ICRA40945.2020.9196557 . hal-02494590

HAL Id: hal-02494590

<https://hal.science/hal-02494590>

Submitted on 28 Feb 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Direct Acceleration Feedback Control of Quadrotor Aerial Vehicles

Mahmoud Hamandi¹, Marco Tognon¹, Antonio Franchi^{2,1}

Abstract—In this paper we propose to control a quadrotor through direct acceleration feedback. The proposed method, while simple in form, alleviates the need for accurate estimation of platform parameters such as mass and propeller effectiveness. In order to use efficaciously the noisy acceleration measurements in direct feedback, we propose a novel regression-based filter that exploits the knowledge on the commanded propeller speeds, and extracts smooth platform acceleration with minimal delay. Our tests show that the controller exhibits a few millimeter error when performing real world tasks with fast changing mass and effectiveness, e.g., in pick and place operation and in turbulent conditions. Finally, we benchmark the direct acceleration controller against the PID strategy and show the clear advantage of using high-frequency and low-latency acceleration measurements directly in the control feedback, especially in the case of low frequency position measurements that are typical for real outdoor conditions.

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have become in the last few years very popular thanks to their versatility and applicability to many different domains. Among the different types of platforms, one of the most popular is the quadrotor. A quadrotor platform can hover in place, which is a desired feature for applications such as inspection, search & rescue, monitoring, etc.

For the purpose of designing the position controller, a quadrotor can be considered as a point mass whose acceleration is the sum of mass-normalized forces such as gravity, wind, and a control force produced by the propellers. The control force can be parameterized in two components 1) its *intensity* which is the total thrust provided by the sum of propeller forces, and 2) its *orientation* which normally corresponds to the vertical axis attached to the vehicle body frame. In its most common form, the motion control of the Center of Mass (CoM) of a quadrotor vehicle boils down to two control loops, the outer and inner loops (see, e.g., [1], [2] and references therein). The outer loop (position control) computes a desired thrust and orientation (roll and pitch) from a blending of the acceleration reference and a feedback based on the CoM position and velocity measurements. The inner loop (attitude control) computes the desired torques to reorient the platform to the desired orientation. Finally, both the desired force and torques are provided to a low

level module that computes rotational speed of each propeller necessary to reach the calculated thrust and orientation.

While each control part is affected by wrongly estimated parameters and external disturbances, this paper focuses on the improvement of the position control, which we shall show to be enough to mitigate disturbances such as uncertain mass and external forces, in addition to uncertain aerodynamic parameters in the low level motor control loop.

To put our controller into perspective, we further divide the position controller into a position feedback loop that computes the desired accelerations based on the measured and desired states, and into a thrust controller that calculates the desired thrust based on the desired accelerations; this second part is commonly implemented in open-loop by applying an inverse of the platform model. To cope with the non-perfect control of the CoM acceleration due to the open-loop nature of the thrust controller, the most common strategy is to make the position feedback loop more robust to input disturbances and parameter uncertainty. Classical solutions resort to different robust controllers (integral position feedback, adaptive control [3], sliding mode [4], MPC [5], etc.), or try to compensate for the inaccurate acceleration control using a disturbance observer [6]–[8].

Another approach that is gaining popularity in the literature is the *Incremental Nonlinear Dynamic Inversion* (INDI) [9]–[11], which exploits the accelerometer measurement to robustify the quadrotor control. The controller is based on an incremental law computed from the Taylor expansion of the dynamics. The robustness of INDI against external disturbances has been proven by real experiments. However, in spite of its robustness, INDI requires the knowledge of the *input effectiveness* (i.e. knowledge of the mass and aerodynamic parameters), and propeller rotational measurements. In addition, INDI low-pass-filters the accelerometer measurements; this introduces a frequency-dependent delay and removes high-frequency components that might be relevant, e.g., in the presence of impacts or rapid changes of acceleration due, e.g., to fast changes in the mass and propeller effectiveness.

In this paper, we propose an alternative solution to the problem; our main idea is rather simple but at the best of our knowledge, it has never been explored in this form in the related literature. Rather than partially relying on the open-loop model inversion, we want to transform the thrust control into a proper feedback control which is based on the direct acceleration measurement. To do so, we exploit the specific acceleration measurement provided by the accelerometer enriching the position controller with an acceleration feedback compensator. The goal is to steer to zero the error between

¹LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France, mahmoud.hamandi@laas.fr, marco.tognon@laas.fr, antonio.franchi@laas.fr,

²Robotics and Mechatronics lab, Faculty of Electrical Engineering, Mathematics & Computer Science, University of Twente, Enschede, The Netherlands a.franchi@utwente.nl

This work was partially funded by the European Union's Horizon 2020 research and innovation programme under grant agreement ID: 871479 AERIAL-CORE

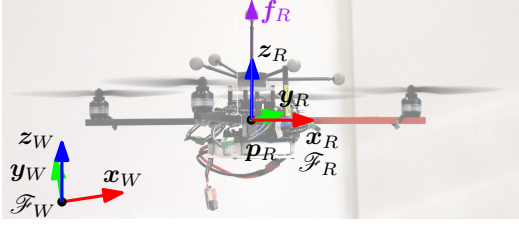


Fig. 1: System model and main variables.

the desired acceleration commanded by the position feedback loop and the measured acceleration coming from the accelerometer. This idea is clearly inspired by torque-feedback-based joint controller for standard manipulators [12], where the electric motor input is commanded in order to steer the measured torque to the desired one. Furthermore, we prove theoretically and experimentally that our controller does not require the knowledge of the mass, propeller rotational speed measurements, and is robust to large variances in the aerodynamic parameters. Finally, to get the most out of the accelerometer measurements, we introduce a zero delay, regression-based multi-notch filter, that leverages upon the commanded propeller speeds to find the signal and noise profiles in the accelerometer measurements.

The paper is organized as follows. In Sec. II we model a generic quadrotor system. In Sec. III we recap the standard outer and inner loop controllers and we present our accelerometer-based inner loop controller, hereafter referred to as the Direct Acceleration (DA) controller. We provide a particular remark showing how the high-frequency acceleration feedback enhances the performance with respect to a PID controller. In Sec. V we present the regression-based filter for the estimation of body acceleration, and finally, Sec. VI shows the experimental results. A conclusive discussion in Sec. VII ends the manuscript and additional experiments and details are provided in the Appendix.

II. MODELING

We firstly define a fixed *world frame*, \mathcal{F}_W , with axes $\{x_W, y_W, z_W\}$ and origin O_W . The position of O_W is arbitrary, while $\{x_W, y_W, z_W\}$ are such that z_W is aligned with the opposite direction of the gravity vector $g \in \mathbb{R}^3$.

To describe the frame of the quadrotor we define a *body frame*, \mathcal{F}_R , with axes $\{x_R, y_R, z_R\}$ and origin O_R centered with the vehicle CoM. We consider a quadrotor aerial vehicle modeled as a rigid body with mass $m_R \in \mathbb{R}_{>0}$ and positive definite inertia matrix $J_R \in \mathbb{R}^{3 \times 3}$ (expressed in \mathcal{F}_R and relative to O_R). The configuration of the vehicle is then given by the position and orientation of \mathcal{F}_R with respect to (w.r.t.) \mathcal{F}_W , described by the vector $p_R \in \mathbb{R}^3$ and the rotation matrix $R_R \in SO(3)$, respectively. The state of the robot completed by the linear velocity w.r.t. \mathcal{F}_R and the angular velocity w.r.t. \mathcal{F}_W , expressed in \mathcal{F}_R , denoted by $v_R \in \mathbb{R}^3$ and $\omega_R \in \mathbb{R}^3$, respectively. On a quadrotor the motion can be controlled acting on four inputs: $f_R \in \mathbb{R}$ and $\tau_R = [\tau_{Rx} \ \tau_{Ry} \ \tau_{Rz}]^T \in \mathbb{R}^3$, where f_R is the magnitude of the thrust force $\mathbf{f}_R = f_R z_R$ applied at O_R and parallel to z_R , and τ_R is the 3D control

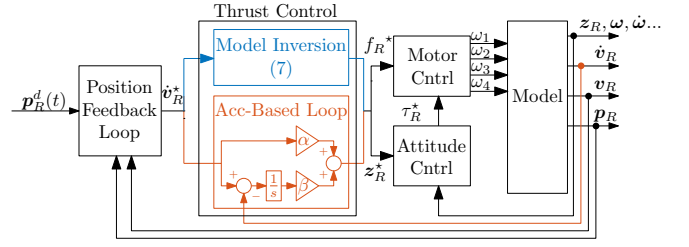


Fig. 2: Block diagram of the quadrotor controller with the two variants of the thrust controller.

moment expressed in \mathcal{F}_R . An example of the system and its main variables is shown in Fig. 1.

Applying the standard Newton-Euler equations one obtains the following vehicle equations of motion:

$$\dot{v}_R = \frac{f_R}{m_R} - g + \frac{f_e}{m_R} \quad (1)$$

$$\dot{R}_R = S(\omega_R) R_R \quad (2)$$

$$J_R \dot{\omega}_R = -S(\omega_R) J_R \omega_R + \tau_R, \quad (3)$$

where $f_e \in \mathbb{R}^3$ is any external force (wind gust, model inaccuracy, aerodynamic effects, etc.). $S(\star)$ is the skew operator defined as $S(\star) : \mathbb{R}^3 \rightarrow \mathbb{R}^{3 \times 3}$ such that $S(x)y = x \times y$ for every $x \in \mathbb{R}^3$ and $y \in \mathbb{R}^3$.

The motion control problem addressed here is to design a feedback law for the control inputs f_R and τ_R that steers the position of the vehicle CoM along a sufficiently smooth desired trajectory $p_R^d(t) : \mathbb{R} \rightarrow \mathbb{R}^3$. It is well known that for a quadrotor, one can also independently control the rotation along the thrust direction, commonly called *yaw angle*. Since the accelerometer cannot improve the control of this quantity, the latter is done using standard techniques [1].

III. CONTROLLER DESIGN

As already discussed, the typical control structure for a quadrotor aerial vehicle is composed of two high level controllers (the position controller and attitude controller) and one low level controller that controls the propeller velocities. In the following, we shall show this standard control law and how we intend to modify the position control part in order to enhance the robustness against unknown parameters and external disturbances, using the direct accelerometer feedback. For the compactness of the paper, we omit the description of the attitude controller that calculates desired torques from desired orientation (and derivatives), and the description of the motor speed allocation; we refer to [1], [2] for the attitude controller and allocation strategy used. Finally, we assume these rotational velocities to be applied instantaneously through a low level controller in the ESC described in [13].

A. Position feedback loop

The position feedback loop is designed considering the dynamics of the vehicle CoM virtually as a double integrator:

$$\dot{p}_R = v_R, \quad \dot{v}_R = \dot{v}_R^*, \quad (4)$$

where $\dot{v}_R^* \in \mathbb{R}^3$ is a controllable virtual input. For this linear system, any stabilizing controller that steers $p_R(t)$ to $p_R^d(t)$

can be used. The simplest implementation is a PD controller:

$$\dot{\mathbf{v}}_R^* = \dot{\mathbf{v}}_R^d + k_P \mathbf{e}_R + k_D \dot{\mathbf{e}}_R, \quad (5)$$

where $\mathbf{e}_R = \mathbf{p}_R^d - \mathbf{p}_R$, and $k_P, k_D \in \mathbb{R}_{>0}$ are positive scalars. We remark that any other more sophisticated controller can be used such as PID, robust controller, sliding mode, etc. However, if (4) is verified, even the simple PD control law (5) will asymptotically steer $\mathbf{p}_R(t)$ to $\mathbf{p}_R^d(t)$.

Then the role of the thrust controller and the attitude controller is to compute the real control inputs f_R and τ_R such that (4) is verified.

B. Classical thrust controller

The classical way to make $\dot{\mathbf{v}}_R = \dot{\mathbf{v}}_R^*$ is to apply a partially open-loop controller based on the inversion of the dynamics. In particular, the thrust vector, \mathbf{f}_R^* , is computed inverting (1):

$$\mathbf{f}_R^* = m_R (\dot{\mathbf{v}}_R^* + \mathbf{g}) - \mathbf{f}_e. \quad (6)$$

As we showed in Sec. II, the total thrust vector is controlled by the thrust intensity and the full-body orientation. Along the current thrust direction, $\mathbf{z}_R^\top \mathbf{f}_R^*$ can be instantaneously applied setting the thrust intensity control input similar to [14]:

$$f_R = \mathbf{z}_R^\top \mathbf{f}_R^*. \quad (7)$$

A schematic representation of the controller along the thrust direction is shown in Fig. 2. In order to properly apply the other two components (lateral forces), it should be that $\mathbf{z}_R = \mathbf{z}_R^*$ where \mathbf{z}_R^* is computed as

$$\mathbf{z}_R^* = \mathbf{f}_R^* / \|\mathbf{f}_R^*\|, \quad (8)$$

where $\|\star\|$ is intended as the *Euclidean norm*. To attain (8), the desired thrust direction, together with the desired rotation along the latter (desired yaw angle), are used as a reference by the attitude controller that steers \mathbf{z}_R to \mathbf{z}_R^* and the yaw angle to the desired one.

In the practical implementation of (7) and (6), one typically uses the nominal values of the system parameters such as the mass \hat{m}_R , the gravity $\hat{\mathbf{g}}$, and an estimation of the external disturbance, $\hat{\mathbf{f}}_e$; concurrently, it is estimated that the lift coefficient applied in the allocation strategy is constant at its nominal value \hat{c}_f . By replacing the nominal values into (6), and placing equation (6) and (7) into (1), we can write the close-loop dynamics along the thrust direction (the most important one for the quadrotor motion control) as follows:

$$\mathbf{z}_R^\top \dot{\mathbf{v}}_R = \mathbf{z}_R^\top \left(\frac{1}{m_R} \left(\mathbf{f}_e - \hat{\mathbf{f}}_e \frac{\hat{c}_f}{c_f} \right) - \mathbf{g} + \frac{\hat{m}_R \hat{c}_f}{m_R c_f} \hat{\mathbf{g}} + \frac{\hat{m}_R \hat{c}_f}{m_R c_f} \dot{\mathbf{v}}_R^* \right) \quad (9)$$

It is clear that, if the nominal and estimated quantities are equal to the real values, i.e., $\hat{m}_R = m_R$, $\hat{\mathbf{g}} = \mathbf{g}$, $\hat{\mathbf{f}}_e = \mathbf{f}_e$ and $\hat{c}_f = c_f$, then $\mathbf{z}_R^\top \dot{\mathbf{v}}_R = \mathbf{z}_R^\top \dot{\mathbf{v}}_R^*$, and the control objective is achieved at least along the thrust direction. Nevertheless, as soon as there are model uncertainties and estimation errors, which is the normal situation in practice, such a goal is not met. In the next section, we shall show how the use of the accelerometer allows to attain the control objective even in those non-ideal conditions.

C. Accelerometer-based thrust controller

The majority of quadrotor platforms are equipped with an Inertial Measurement Unit (IMU) including an accelerometer and a gyroscope. The two are normally used for the estimation of the vehicle attitude [1], [15]. Here we shall show how the acceleration measurement can also be used to make the (partially open) position loop controller a (fully) closed-loop one. This allows achieving the control objective (4) even with model uncertainties and unknown external disturbances.

The accelerometer measurements, defined by the vector $\mathbf{w}_a \in \mathbb{R}^3$, provide the specific acceleration, i.e.,

$$\mathbf{w}_a = \mathbf{R}_R^\top (\dot{\mathbf{v}}_R - \mathbf{g}), \quad (10)$$

from which one can get a direct measure of the acceleration

$$\dot{\mathbf{v}}_R = \mathbf{g} + \mathbf{R}_R \mathbf{w}_a. \quad (11)$$

As done in the previous section let us focus on the dynamics along the thrust direction. To simplify the notation we define $y = \mathbf{z}_R^\top \dot{\mathbf{v}}_R$, $u = \mathbf{z}_R^\top \mathbf{f}_R = f_R$, $a = (c_f / (m_R \hat{c}_f))$ and $b = \mathbf{z}_R^\top (\mathbf{f}_e / m_R - \mathbf{g})$. We can then rewrite (1) along the thrust direction as

$$y = au + b. \quad (12)$$

We consider a and b as unknown variables. More in general, the quantity a , also called *input effectiveness*, includes different sources of uncertainties: 1) the mass and 2) the thrust intensity which might be not precisely known. The latter might even change over time due to aerodynamic effects, e.g., ground and ceiling effects [16]. The quantity b represents any external disturbance applied to the robot (e.g., wind gusts, the additional weight of a load, unmodeled drag, etc.), including the gravity. The control objective is to steer y to $y^* = \mathbf{z}_R^\top \dot{\mathbf{v}}_R^*$ with a and b unknown. To attain this purpose we propose the following controller based on the acceleration input

$$u = \alpha y^* + \beta \int e_y dt, \quad (13)$$

where $e_y = y^* - y$ is the acceleration error, $\beta \in \mathbb{R}$, and $\alpha \in \mathbb{R}$ is an estimation of $1/a$. We shall show that the term αy^* improve the control performance if α is a decent approximation of $1/a$. However, if a good estimation of a is not available, α can be set arbitrarily and the controller is still applicable. A schematic representation of the controller along the thrust direction is shown in Fig. 2. We can verify that e_y has a stable behavior analyzing its dynamics:

$$\dot{e}_y = y^* - (a\dot{u} + \dot{a}u + \dot{b}) = -a\beta e_y + \tilde{a}y^* - \dot{a}u - \dot{b}, \quad (14)$$

where $\tilde{a} = 1 - a\alpha$. The equilibrium point is clearly $\bar{e}_y = (\tilde{a}y^* - \dot{a}u - \dot{b}) / (a\beta)$ and is asymptotically stable if $a\beta > 0$, while assuming \dot{a} to be zero. Even if a is unknown, its sign is normally known and the sign of β can be chosen accordingly. Without loss of generality, we can assume that $a > 0$. The equilibrium point \bar{e}_y is then asymptotically stable for any $\beta > 0$, a and b .

Remark 1. Notice that if $y^* = \dot{a} = \dot{b} = 0$, as in many of the practical cases, the error will asymptotically converge to zero. One can also observe that knowing precisely the input

effectiveness, (i.e., $\alpha = 1/a$, $\tilde{a} = 0$) the part of the error due to \dot{y}^* vanishes. It is not surprising that knowing the feed-forward term will improve the performance. Nevertheless we remark that for the DA controller a precise knowledge of a is not actually needed. In fact, increasing β , one can make the steady-state error \bar{e}_y very small, independently of the unknown parameter and disturbance. Unfortunately, β cannot be arbitrarily large otherwise \dot{u} will be too large and the system will not be able to provide the corresponding u .

The above remark is validated experimentally¹.

IV. DISCUSSION ON THE COMPARISON WITH PID

Let us assume that a standard PD position feedback loop controller is applied. We can then replace (5) into (13), where we recall that $y^* = z_R^\top \dot{v}_R^*$:

$$u = \alpha \left(\dot{v}_{Rt}^d + k_P e_{Rt} + k_D \dot{e}_{Rt} \right) + \beta \left(\dot{e}_{Rt}' + k_D \dot{e}_{Rt}' + k_P \int e_{Rt} dt \right). \quad (15)$$

where $\dot{v}_{Rt}^d = z_R^\top \dot{v}_R^d$, $e_{Rt} = z_R^\top e_R$ and similarly for its derivatives. Furthermore, $\dot{e}_{Rt}' = \int (\dot{v}_{Rt}^d - v_{Rt}) dt$ and $\dot{e}_{Rt}' = \int (\dot{v}_{Rt}^d - \dot{v}_{Rt}) dt$ with $v_{Rt} = z_R^\top v_R$ and similarly for its derivatives. While it is clear that (15) resembles a PID, it turns out from the experimental comparisons (see Fig. 5) that the step response against an external disturbance can be made faster for the proposed acceleration-based thrust controller plus a PD position feedback loop controller when compared to the PID controller. The experiment has been conducted with the two controller gains tuned at best².

To investigate this discrepancy, we notice that the PID in (15) relies on the high frequency accelerometer measurements 1 [KHz], in addition to the low frequency position and velocity measurements required in a normal PID.

In the following, we analyze a simplified system that still encapsulates the main properties, in order to better explain why high-frequency acceleration feedback shows better stability than an equivalent controller based on slower sampled measurements. Let us consider the dynamic system

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = u, \quad (16)$$

where $u = -k_1 x_1 - k_2 x_2$ is a simple PD controller with $k_1, k_2 \in \mathbb{R}_{>0}$. For this system we analyze two cases:

- 1) both x_1 and x_2 in u are sampled with period $T \in \mathbb{R}_{>0}$;
- 2) only x_1 in u is sampled with period T , while x_2 is sampled at a much higher frequency such that it can be considered continuous.

In our parallelism, case 1) corresponds to the standard PID based on low-sampled measurements, while case 2) corresponds to the proposed acceleration-based thrust controller.

For case 1), from the theory of digital control [17], it is well known that the proportional gain k_1 cannot be increased arbitrarily (aiming at a better performance). On the contrary,

¹The results and details of these experiments, showing the robustness to the choice of a and the advantage of knowing accurately the input effectiveness, are presented in the Appendix.

²We consider the best gains as the maximum gains that preserve the stability of the system. Higher gains would make the system unstable.

its maximum value is bounded depending on the sampling time T . The higher T the lower k_1 has to be in order to guarantee stability.

For case 2) the measurement of x_2 can be considered continuous time with respect to the sampling of x_1 . Then, the control input u can be written as

$$u = -k_1 \bar{x}_1 - k_2 x_2, \quad (17)$$

where \bar{x}_1 is the sampled measurement of x_1 . In the continuous-time we can write $\bar{x}_1(t) = x_1(t) - \Delta x_1(t)$ where $\Delta x_1(t) \in \mathbb{R}$ is the error at time t due to the sampling of the signal. Notice that $\Delta x_1(t) = 0$ if $t = iT$ for a certain i . Let us now consider the following Lyapunov function:

$$V(x_1, x_2) = \frac{1}{2} (k_1 x_1^2 + x_2^2), \quad (18)$$

which is clearly positive semi-definite, and $V(x_1, x_2) = 0$ if $x_1 = x_2 = 0$. The time derivative of $V(x_1, x_2)$ is

$$\begin{aligned} \dot{V}(x_1, x_2) &= k_1 x_1 x_2 + x_2 (-k_1 \bar{x}_1 - k_2 x_2) \\ &= x_2 (k_1 \Delta x_1 - k_2 x_2). \end{aligned} \quad (19)$$

Considering the Taylor approximation of Δx_1 around the time iT for a certain i , we can bound Δx_1 with $x_2 T$. Therefore, it is easy to verify that

$$\dot{V}(x_1, x_2) \leq -x_2 (k_2 - k_1 T) x_2 \quad (20)$$

which is negative semi-definite if k_2 is chosen such that $k_2 - k_1 T > 0$. Using the LaSalle principle [18], we can prove the stability of the system. This implies that, whatever is the sampling period of x_1 , one can choose k_1 arbitrarily large, and then k_2 such that $k_2 > k_1 T$. For case 2), thanks to the high sampling frequency of x_2 , one can obtain a fast convergence of the state to zero independently from the sampling rate of x_1 .

V. FILTERING OF THE BODY ACCELERATION

The raw accelerometer measurements are practically unusable due to high frequency noise, mostly coming from the vibrations of the propellers. The most common technique to extract the body acceleration is to use a low-pass filter, which introduces frequency dependent delay and phase shift, and can remove high frequency components of the signal. Moreover, low-pass filters are linear filters that attenuate high frequency noise, instead of removing them. As such, high amplitude noise can still show up in the filtered signal.

We designed an adaptive and regression-based notch filter that allows to reject multiple frequency components, without adding any delay or phase shift, and that at the same time remains simple to implement. In addition, as our filter is regression-based, it is able to find the amplitude of the noise signal and remove it completely from the accelerometer signal while a linear filter can only attenuating it.

Considering what has been presented so far, we can modify the ideal accelerometer model in (10) into:

$$w_a = {}^S \dot{v}_R + \delta + \sigma, \quad (21)$$

where ${}^S \dot{v}_R = R_R^\top (\dot{v}_R - g)$ is the specific acceleration of the main body, the one that we would like to estimate, δ represents the vibrations induced by spinning propellers,

and σ represents all the additional noise, assumed gaussian distributed with zero mean.

Let us consider a certain time window of $N \in \mathbb{N}_{>0}$ acceleration samples, $\tilde{w}_a(t) = [w_a^T(t) \ w_a^T(t-T) \ \dots \ w_a^T(t-NT)]^T \in \mathbb{R}^{3N}$, where $T \in \mathbb{R}_{>0}$ is the sampling period. The first step to estimate ${}^S\dot{v}_R(t)$ given $\tilde{w}_a(t)$ is to model the components of w_a . For the specific acceleration we use a Taylor/polynomial approximation:

$${}^S\dot{v}_R(t) = \sum_{i=0}^p \theta_{vi} t^i, \quad (22)$$

where $p \in \mathbb{N}_{>0}$ is the degree of the polynomial which has to be chosen a priori and depends on the variability signal ${}^S\dot{v}_R$ over the time window $[t-NT, t]$ (as a rule of thumb, in general, the larger the time window the larger p), and $\theta_{vi} \in \mathbb{R}$ is the generic unknown i -th coefficient of the polynomial to be estimated.

We model δ as a sum of q shifted sinusoidal functions:

$$\delta(t) = \sum_{j=0}^q \theta_{\delta j} \sin(\omega_{\delta j} t + \phi_{\delta j}), \quad (23)$$

where $\theta_{\delta j} \in \mathbb{R}$ and $\phi_{\delta j} \in \mathbb{R}$ are the amplitude and phase shift of the generic j -th sinusoidal, again to be estimated. We assume that the pulsation $\omega_{\delta j}$ of the j -th sinusoidal is known since it is directly linked to the spinning frequency of the propellers and its harmonics. There are several ways to choose q and $\omega_{\delta j}$ according to the current frequency of the propellers. In general, it is a good practice to choose q as small as possible to avoid over-fitting. Following this guideline, we choose $q = 4$ taking $\omega_{\delta 1}$ and $\omega_{\delta 2}$ as the minimum spinning frequency among the propellers (in the considered time window) and its first harmonic, respectively. Similarly, we take $\omega_{\delta 3}$ and $\omega_{\delta 4}$ as the maximum spinning frequency among the propellers (in the considered time window) and its first harmonic, respectively.

Noticing that $\sin(\omega_{\delta j} t + \phi_{\delta j}) = \cos(\phi_{\delta j}) \sin(\omega_{\delta j} t) + \sin(\phi_{\delta j}) \cos(\omega_{\delta j} t)$ we can rewrite (23) as

$$\delta(t) = \sum_{i=0}^q \theta_{ci} \sin(\omega_{\delta i} t) + \theta_{si} \cos(\omega_{\delta i} t), \quad (24)$$

which is now linear in the new parameters $\theta_{ci} = \theta_{\delta j} \cos(\phi_{\delta j})$ and $\theta_{si} = \theta_{\delta j} \sin(\phi_{\delta j})$.

Given the accelerometer measurements \tilde{w}_a , we can obtain an estimation of the specific acceleration, ${}^S\hat{v}_R$, identifying the parameters θ_{vi} for $i = 1, \dots, p$, and θ_{sj}, θ_{cj} for $j = 1, \dots, q$, by simple linear regression. Notice that the minimum requirement is $N \geq p + 2q$. In addition, in order to clearly identify all the sinusoidal components, it is desirable that \tilde{w}_a contains at least a full period of the slowest sinusoidal signal.

Finally, in order to remove the additional noise σ (that we did not model in the regression) from ${}^S\hat{v}_R$ we use a discrete adaptive high-bandwidth low-pass filter. Let ${}^S\hat{v}_R[k]$ be the regressed estimation of the specific acceleration at the discrete-time $t_k = kT$, where $k \in \mathbb{N}^+$. Then the adaptive high-bandwidth low-pass filter can be expressed as:

$${}^S\hat{v}_{Rf}[k] = (1 - \kappa[k]) {}^S\hat{v}_{Rf}[k-1] + \kappa[k] {}^S\hat{v}_R[k], \quad (25)$$

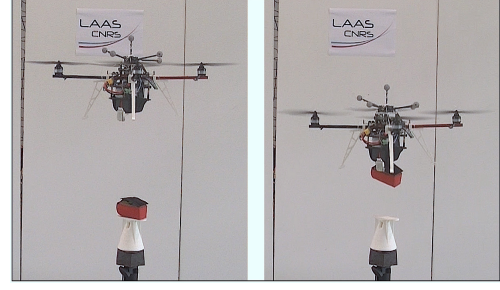


Fig. 3: The quadrotor picking up an unknown mass with a hook-and-loop. Left: the platform descending to pick up the object. Right: the platform ascending after it had fastened and picked up the object.

where ${}^S\hat{v}_{Rf}[k]$ is the low-pass filtered value of ${}^S\hat{v}_R[k]$, and $\kappa[k]$ is an adaptive gain computed as:

$$\kappa[k] = \min \left(1, \left| \frac{{}^S\hat{v}_R[k]}{\dot{v}^{th}} \right| \right), \quad (26)$$

where $|\cdot|$ represents the absolute value, ${}^S\hat{v}_R[k]$ is an estimation of the rate of change of ${}^S\hat{v}_R$ calculated from the regressed derivative of ${}^S\hat{v}_R$, and \dot{v}^{th} is a threshold rate such that if ${}^S\hat{v}_R[k] \geq \dot{v}^{th}$ the filter would behave as a zero-phase all-pass filter.

The chosen adaptive gain allows the filter to have low bandwidth when ${}^S\dot{v}_R$ is slowly varying (a case in which the effect of σ is more visible), thus effectively filtering the noise, and a very high bandwidth when ${}^S\dot{v}_R$ is rapidly varying (a case in which the effect of σ is almost negligible) to obtain non-delayed tracking³.

VI. EXPERIMENTAL RESULTS

The validation experiments have been conducted with a quadrotor platform, with software running on-board in real-time. The experiments are conducted in door with conditions emulating outdoor settings. For more details about the hardware, software and configurations used, we refer the reader to the Appendix.

A. Use of DA in Two Practically Relevant Scenarios

In this section we assess the performance of the proposed DA controller while completing two real world tasks involving unknown and time-varying changes in the quadrotor dynamical model: *i*) hook-and-loop fasten and pick an unknown-mass object, and *ii*) take-off and maneuver in a air-turbulent environment⁴

In the first experiment, shown in Fig. 3, the platform is commanded to pick up an object whose 0.4 [Kg] mass is unknown to the controller. Despite the unknown mass, the platform was able to follow the desired position while approaching and lifting the object. The recorded error along both trajectories is less than 0.008 [m].

In the second experiment, shown in Fig. 4, the platform flies underneath a flying hexarotor; due to the configuration

³The proposed filter is compared against a second-order Butterworth filter; the comparison results are presented in the Appendix.

⁴Detailed results of these experiments are presented in the Appendix.



Fig. 4: The quadrotor flying stably beneath a 2.5 [Kg] hexarotor.

of its propellers, the hexarotor creates a turbulent environment for the quadrotor. Despite the turbulence, the quadrotor is able to lift off ground, hover in place, then approach the hexarotor and retract back to its original position with an error less than 0.01 [m] throughout its flight.

B. Comparing the Best DA and PID Performances

We assess the performance of DA against a PID controller with equivalent gains as described in (15). Gains k_P and k_D were tuned to ensure a desirable transient performance, and a near zero steady state error: $k_P = 27.5$ and $k_D = 18$. The acceleration gain β was varied to study its effect on the performance of each controller; as seen from (13), the increase in β increases the reactivity of the controller to e_y , and correspondingly its reactivity to external disturbances.

Figure 5 shows a statistical representation of repeated experiments demonstrating a comparison between the DA controller and a PID following a step disturbance of 4 [N] – equivalent to $\simeq 0.407$ [Kg], i.e., 40% of the platform mass. This experiment highlights the performance of the controllers in scenarios such as sudden lifting of an unknown mass, or a very quick change in the aerodynamic properties of the platform's actuators, due, e.g, to a sudden wind gust. The same figure shows the difference between the controllers for different values of the acceleration gain β .

We observe that both controllers can achieve a zero steady state error before and after the step disturbance, however with varying transient performances. As we increase β , both controllers reach steady state in a shorter time, and achieve a smaller maximum error post the step response; these last two observations suggest that an increase in β induces a faster reaction to external disturbances.

We can also observe from Fig. 5 that the performance of both controllers is similar for the same β , with the PID lagging slightly from the DA controller. However, we can see that at $\beta = 4$ the PID controller lets the error oscillate and eventually diverge, while the DA controller can still achieve stable flight. At $\beta = 12$ the DA controller shows small oscillations – while still achieving a stable flight – with a maximum error of 2 [cm], while the PID's best error was around 4 [cm], with $\beta = 3$.

C. Dynamic Maneuver

Finally, we assess the performance of DA while per-

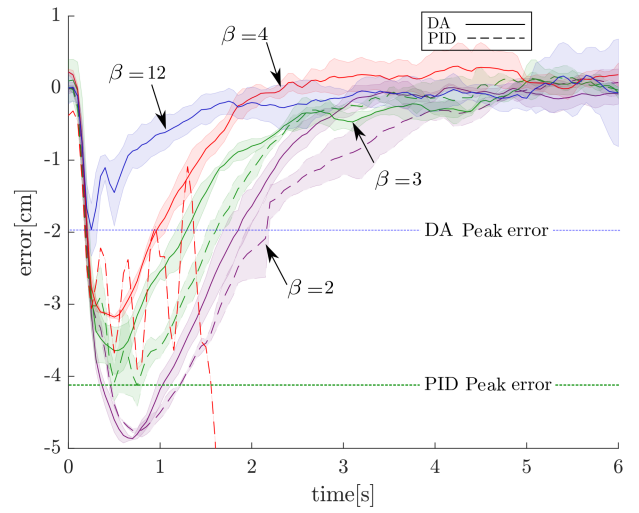


Fig. 5: Step response of DA vs PID following a 4 [N] ($\simeq 0.407$ [Kg]) step function while varying β , with lines of the same color corresponding to the same β , solid lines to DA, and dashed lines to PID. Each of the plots show the statistics of 3 identical flights, with the exception of the PID at $\beta = 4$ showing only one flight to avoid possible crashes.

forming high speed maneuvers. While our system does not ameliorate the attitude controller, but rather is designed to be robust to external disturbances, we conduct these flights to show that the lateral motion is not deteriorated by our controller.

To this end, we fly our controller in a lemniscate pattern, with a radius of 2 [m] and at a maximum speed of 2.8 [m.s⁻¹]. During these flights, our controller follows the desired state defined by the equation of the lemniscate with the corresponding derivatives. The controller exhibits a maximum error of 0.1 [m] along the trajectory, which is equivalent to the PID error for the same flight. More details regarding this flight can be found in the Appendix.

VII. CONCLUSIONS

In this work we proposed a novel quadrotor controller concept based on direct acceleration (DA) feedback. The controller does not require any knowledge of the platform's mass or propeller efficiency, while can still benefit from such knowledge.

In addition, we proposed a new regression-based method to filter the IMU signal which uses the information about the commanded propeller speed, and extracts *vibration-free* body acceleration. We show that our regression based method removes the vibrations from the signal with a smaller delay than classical low pass filters.

We conducted an experimental campaign to demonstrate the performance of the new controller in various challenging scenarios, and compared its performance to a well tuned PID. During these experiments, the DA controller exhibits more accurate position tracking conditions in reaction to sudden disturbances.

In the future we intend to improve the DA controller by including possible compensation also of the lateral acceleration

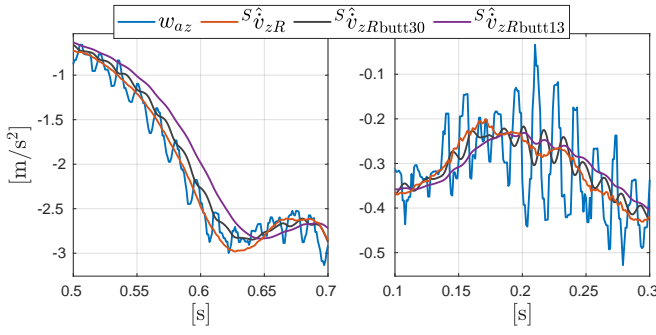


Fig. 6: Comparison of different types of filters on the raw accelerometer data w_{az} , where \hat{v}_R is the output of the proposed regression-based filter, $\hat{v}_{R\text{butt}30}$ is the output of a second-order Butterworth filter with cutoff frequency 30 [Hz], and $\hat{v}_{R\text{butt}13}$ is the output of a second-order Butterworth filter with cutoff frequency 13 [Hz].

and the handling of substantial biases in the measurement data by including pre-integration of the positional error, in addition to the existing post-integration.

APPENDIX

This appendix presents details the experimental setup, and extra experimental results.

A. Regression Based notch filter

In Fig. 6 we compared the estimated specific acceleration using different techniques: 1) the proposed regression-based filter, defined by the symbol \hat{v}_R ; 2) a second-order Butterworth filter with cutoff frequency 30 [Hz] (same setup used in [9]), defined by the symbol $\hat{v}_{R\text{butt}30}$; 3) a second-order Butterworth filter with cutoff frequency 13 [Hz], defined by the symbol $\hat{v}_{R\text{butt}13}$. From Fig. 6 it is evident that the proposed regression-based filter is the one able to suppress the vibrations (see Fig. 6 on the right) while minimizing the delay when the signal changes rapidly (see Fig. 6 on the left). On the other hand, using a second-order Butterworth filter with cutoff frequency 30 [Hz], the delay with respect to the real signal is small (although still larger than using the regression-based filter) but the vibrations are still there (see Fig. 6 on the right). In order to almost suppress the vibrations with a second-order Butterworth filter, one has to set a cutoff frequency of 13 [Hz]. However the delay becomes much larger.

B. Experimental Setup

The validation experiments have been conducted with a quadrotor platform, weighting about 1 [Kg]. The vehicle is endowed with an IMU, which exports the raw accelerometer and gyroscope measurements at 1 [KHz], and four brushless motor controllers (BLDC ESC) regulating the propeller speed using an in-house developed closed-loop speed controller [13]. A motion capture system reads the position and orientation of the vehicle and a UKF component performs the sensor fusion to retrieve the full state of the platform. To ensure consistency with the typical sensor frequencies available in an outdoor setting endowed

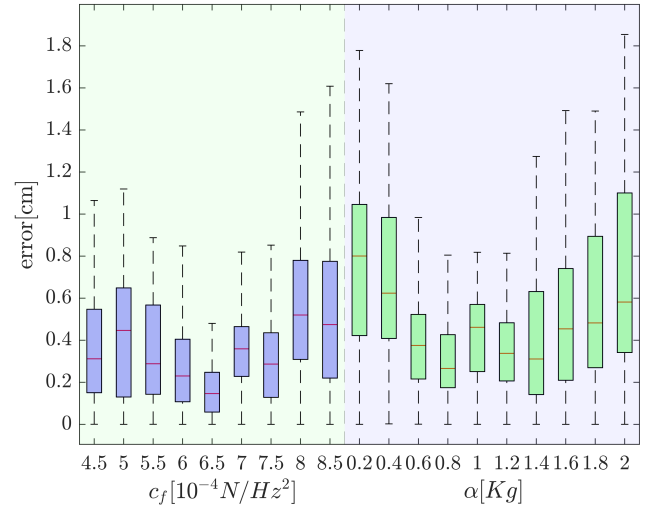


Fig. 7: showing error distribution of the platform while taking off and landing with a wrong c_f or α .

with, e.g., a standard vision-based localization system, we sample the position and velocity measurements from the UKF at 20 [Hz]. Most of the software components (including the controller) have been developed in C++ and run on an onboard PC (odroid XU4) at 1 [KHz]. As for most of the robotics software at LAAS-CNRS, software components have been developed using GenoM3 [19], a code generator and formal software component description language that allows assembling middleware-independent components in a modular system. Most of this software is available on the *openrobots* repository at <https://git.openrobots.org/projects/telekyb3>

C. Flight with unknown parameters

This section shows experiments where the platform takes off and lands with different values of the lift coefficient c_f and α provided to the attitude controller and the thrust controller respectively.

- In the first flight we vary $\alpha \in [0.2, 2][Kg]$, while its nominal value is $\frac{1}{a} = 1 [Kg]$.
- In the second flight we vary the estimated lift coefficient $c_f \in [4.5e^{-4}, 8.5e^{-4}][N/Hz^2]$, while its identified nominal value is $\hat{c}_f = 6.5e^{-4}[N/Hz^2]$.

Fig. 7 shows the distribution of the platform's error while following the above maneuver with the chosen parameters. The figure shows that the platform is able to fly for the given α and c_f range. We note that the tested c_f range is one that might be encountered during regular flights (such as ground effect [20]), however, we acknowledge that our controller does not guarantee stability for larger variances in c_f due to the presence of the attitude controller. In addition, we note from Fig. 7 that the performance improves if α is an exact estimate of $\frac{1}{a}$, however, the controller is still stable even when α is set arbitrarily as explained earlier.

D. Use of DA in Two Practically Relevant Scenarios

Fig. 8 shows the actual-vs-desired quadrotor altitudes before and after hook-and-loop fastening and picking up an

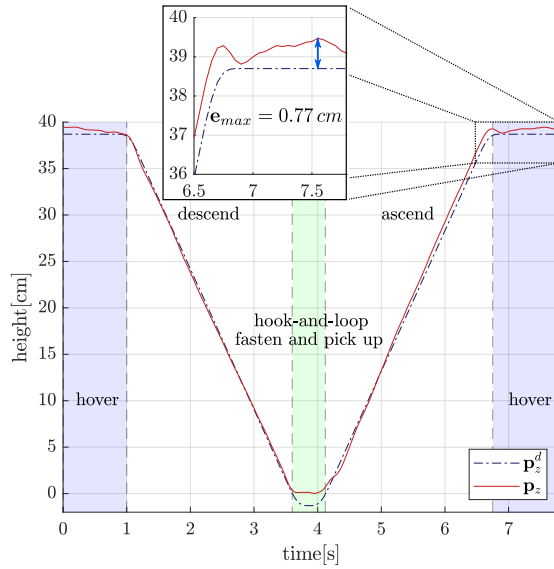


Fig. 8: position tracking of the quadrotor before and after hook-and-loop fasten and pick up of an object whose 0.2 [Kg] mass is unknown to the controller.

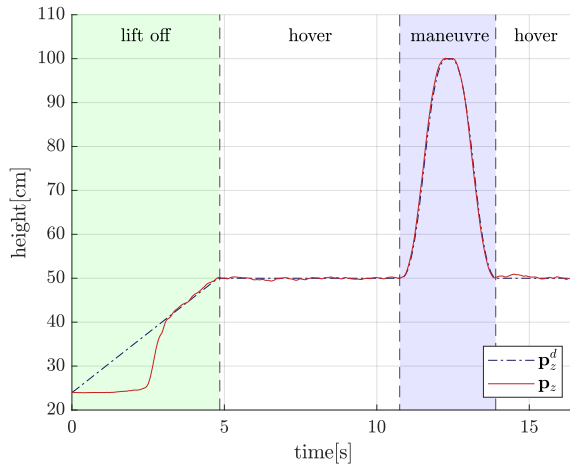


Fig. 9: position tracking of the quadrotor flying beneath a 2.5 [Kg] hexarotor. The plot shows a lift off phase, followed by a hover phase, a manoeuvre phase, and finally another hover phase. During the first 2.5 [s], the platform follows a safety procedure imposing a ramp on the motor speeds, irrespective of the controller commands.

object whose 0.4 [Kg] mass is unknown to the controller. The top surface of the object is taken as a reference point. The platform is commanded to go below the object surface as seen in the hook-and-loop fastening and pick up section of Fig. 8. This experiment shows that while the weight of the object is unknown to the controller, and there is a phase in which the motion is even hindered, the platform's performance is smooth, stable, and indistinguishable before and after the pick up, with less than 0.008 [m] error in both phases.

Fig. 9 shows the performance of the DA in the presence of turbulence created by flying near ground and underneath a flying hexarotor weighting about 2.5 [Kg]. The platform first lifts off from ground and hovers in place, then it increases its

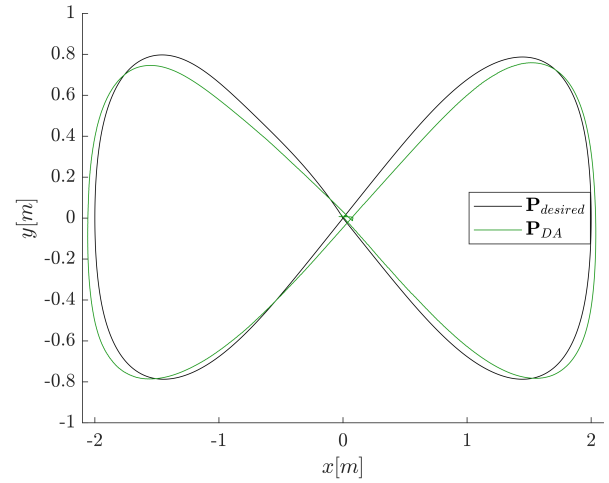


Fig. 10: position tracking of the quadrotor while following a high speed lemniscate manoeuvre.

altitude to approach the bottom of the hexarotor, and then it goes back to hover in the previous altitude. The DA controller in Fig. 9 shows complete resilience to the existing turbulence while following its desired trajectory, with less than 0.01 [m] maximum error. It has to be noted that during take off there is an initial error caused by the quadrotor's safety starting procedure which imposes the motor speeds to follow a 2.5 [s] saturation ramp from 0% to 100% which prevents the execution of the DA commanded forces. The DA controller lets the platform behave smoothly and stably despite such additional unmodeled obstruction to the actuation system.

E. Lemniscate Manoeuvre

Fig. 10 shows the performance of the DA controller while following an agile lemniscate manoeuvre. The lemniscate has a 2 [m] radius, and with a maximum velocity in the center at 2.8 [m.s⁻¹], and 2 [m.s⁻¹] with a 2.8 [m.s⁻²] acceleration at the edges. The lemniscate is enforced by calculating the desired position, velocity, acceleration, jerk and snap at each point.

The maximum error norm recorded along the trajectory is 0.1 [m].

REFERENCES

- [1] R. Mahony, V. Kumar, and P. Corke, "Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.
- [2] M. Faessler, A. Franchi, and D. Scaramuzza, "Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 620–626, 2018.
- [3] G. Antonelli, E. Cataldi, F. Arrichiello, P. Robuffo Giordano, S. Chiverini, and A. Franchi, "Adaptive trajectory tracking for quadrotor MAVs in presence of parameter uncertainties and external disturbances," *IEEE Trans. on Control Systems Technology*, vol. 26, no. 1, pp. 248–254, 2018.
- [4] L. Besnard, Y. B. Shtessel, and B. Landrum, "Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer," *Journal of the Franklin Institute*, vol. 349, no. 2, pp. 658–684, 2012.

- [5] K. Alexis, G. Nikolakopoulos, and A. Tzes, "Switching model predictive attitude control for a quadrotor helicopter subject to atmospheric disturbances," *Control Engineering Practice*, vol. 19, no. 10, pp. 1195–1207, 2011.
- [6] B. Yüksel, C. Secchi, H. H. Bühlhoff, and A. Franchi, "Aerial physical interaction via IDA-PBC," *The International Journal of Robotics Research*, vol. 38, no. 4, p. 403421, 2019.
- [7] M. Ryll, G. Muscio, F. Pierri, E. Cataldi, G. Antonelli, F. Caccavale, D. Bicego, and A. Franchi, "6D interaction control with aerial robots: The flying end-effector paradigm," *The International Journal of Robotics Research*, vol. 38, no. 9, pp. 1045–1062, 2019.
- [8] T. Tomić, C. Ott, and S. Haddadin, "External wrench estimation, collision detection, and reflex reaction for flying robots," *IEEE Trans. on Robotics*, vol. 33, no. 6, pp. 1467–1482, 2017.
- [9] E. Tal and S. Karaman, "Accurate tracking of aggressive quadrotor trajectories using incremental nonlinear dynamic inversion and differential flatness," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 4282–4288.
- [10] E. J. Smeur, Q. Chu, and G. C. de Croon, "Adaptive incremental nonlinear dynamic inversion for attitude control of micro air vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 12, pp. 450–461, 2015.
- [11] E. J. Smeur, G. C. de Croon, and Q. Chu, "Cascaded incremental nonlinear dynamic inversion for mav disturbance rejection," *Control Engineering Practice*, vol. 73, pp. 79–90, 2018.
- [12] D. Vischer and O. Khatib, "Design and development of high-performance torque-controlled joints," *IEEE Trans. on Robotics and Automation*, vol. 11, no. 4, pp. 537–544, 1995.
- [13] A. Franchi and A. Mallet, "Adaptive closed-loop speed control of BLDC motors with applications to multi-rotor aerial vehicles," in *2017 IEEE Int. Conf. on Robotics and Automation*, Singapore, May 2017, pp. 5203–5208.
- [14] T. Lee, M. Leoky, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," in *49th IEEE Conf. on Decision and Control*, Atlanta, GA, Dec. 2010, pp. 5420–5425.
- [15] P. Martin and E. Salaün, "The true role of accelerometer feedback in quadrotor control," in *2010 IEEE Int. Conf. on Robotics and Automation*, Anchorage, AK, May 2010, pp. 1623–1629.
- [16] C. Powers, D. Mellinger, A. Kushleyev, B. Kothmann, and V. Kumar, "Influence of aerodynamics and proximity effects in quadrotor flight," in *Experimental robotics*. Springer, 2013, pp. 289–302.
- [17] I. D. Landau and G. Zito, *Digital control systems: design, identification and implementation*. Springer Science & Business Media, 2007.
- [18] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2001.
- [19] A. Mallet, C. Pasteur, M. Herrb, S. Lemaignan, and F. Ingrand, "Genom3: Building middleware-independent robotic components," *2010 IEEE International Conference on Robotics and Automation*, pp. 4627–4632, 2010.
- [20] D. D. C. Bernard, M. Giurato, F. Riccardi, and M. Lovera, "Ground effect analysis for a quadrotor platform," in *Advances in Aerospace Guidance, Navigation and Control*. Springer, 2018, pp. 351–367.