2022 IEEE Conference on Control Technology and Applications (CCTA)
August 22-25, 2022. Trieste, Italy

Drone reference tracking in a non-inertial frame using sliding mode control based Kalman filter with unknown input

Yasmine Marani[1], Kuat Telegenov[1], Eric Feron[1], Meriem-Taous Laleg Kirati[1,2]

***Abstract*— As surprising as it seems, drones and mobile robots in general experience motion sickness when put in a moving environment. This navigation problem has been little if ever explored in the literature. Therefore we propose a formulation of the problem in the simplest possible way as a starting point. The objective of simplifying the problem is to avoid using sophisticated control and measurement devices, such as cameras, and rely instead on control system strategies. In this paper, the moving environment to which is associated a non-inertial frame is considered to have translation motion with respect to the inertial reference frame. The goal is to make the drone track a desired trajectory inside the moving environment based only on the measurements obtained with respect to the non-inertial frame. First, a model representing the dynamics of the drone in the non-inertial frame is developed using the relative motion principles. The new model takes into account the accelerations of the moving environment where they are considered as bounded unknown inputs. Then, a Kalman Filter with Unknown Inputs (KF-UI) is used to estimate simultaneously the states of the drone and the accelerations of the non-inertial frame. Finally, a Sliding Mode controller is implemented. Two numerical simulations were conducted to illustrate the performance of the combined KF-UI and Sliding Mode controller: the first one represents an ideal case where the non-inertial frame's accelerations are constant. The second one illustrates flying a drone in an elevator. The obtained results form an encouraging foundation for follow-on experiments.**

I. INTRODUCTION

Over the past decade, Unmanned aerial vehicles (UAVs), also known as drones, have become a popular tool for many various applications, such as aerial photography, search and rescue, precision agriculture, pipeline/overhead transmission-line monitoring, aerial mapping, wildlife monitoring, surveillance, disaster management, and delivery of medical supplies [1]–[18]. For these applications, Global Navigation Satellite System (GNSS) or instrumented positioning infrastructures are required. Nevertheless, UAVs may show usefulness for many other undiscovered applications in GPS-denied moving environments such as cars, boats, trucks, trains, ships, airplanes, and elevators. However, flying in these environments can be challenging for a variety of reasons. Having no GPS signal deprives the UAV of direct access to absolute position in the Geocentric coordinate system, which, for vehicles like drones, is considered to be an inertial frame. The onboard accelerometer and gyroscope will provide angular rates and acceleration with respect to an inertial frame. In contrast, the onboard localization sensors will provide a position with respect to the moving environment, non-inertial frame. As a result, UAV visual odometry will be dissonant with its inertial measurements. Several Youtube videos show people attempting to fly UAVs inside trucks or elevators as an illustration of this challenge [19]–[23]. The results achieved by human pilots at very low speeds are decent at best, but inconsistent. Nevertheless, as the vehicle acceleration becomes larger, it becomes more difficult or near impossible to control the UAV. There have been few works related to UAVs operating with non-inertial reference frames. In [24] authors present an autonomous landing control approach for a quadrotor unmanned aerial vehicle subject to wind disturbance and three-dimensional movements of the landing platform. Yet another work addresses the problem of controlling tethered aerial vehicle by considering the aerial robot linked to a generic and independently moving platform [25]. Another similar work where authors address the tracking issue for a non-inertial frame referenced quadrotor Unmanned Aerial Vehicle which is controlled by a cascaded PID controller [26]. None of these works considered flying UAVs inside the non-inertial environment or platform without access to inertial position information. To the best of the authors' knowledge, autonomous UAVs flying inside of the moving environment with no inertial positioning of sorts have not been studied or presented in the literature. As an initial attempt to get a better sense of the problem, the present paper provides a proper formulation in which the moving environment is assumed to have a translation motion with respect to the geostationary reference frame.

The modeling of drones dynamics has been widely investigated in the literature. Three modeling approaches are used: first principle modeling, grey box-modeling, and black-box modeling. However, the first modeling approach is the one that provides the most physical insights. In [27]–[29] a non-linear model was developed based on the Newton-Euler formalism, and in [30] a hybrid dynamic model was derived. All these models were written with respect to an inertial reference frame. However, we consider in this work that the drone is inside a moving environment which is associated with a non-inertial frame, and does not have access to measurement in the inertial frame. Therefore, a new model of the quadrotor in the non-inertial frame is developed using the relative motion principle. The developed model consists primarily of the nonlinear model in [28] and combines it with the acceleration of the moving environment. This acceleration acts like an unknown input in the dynamic

[1]Computer, Electrical and Mathematical Science and Engineering Division (CEMSE), King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia. [2]Taous Meriem Laleg-Kirati is also affiliated to the National Institute for Research in Digital Science and Technology, Paris-Saclay, France. yasmine.marani@kaust.edu.sa, kuat.telegenov@kaust.edu.sa, eric.feron@kaust.edu.sa, taousmeriem.laleg@kaust.edu.sa

2022 IEEE Conference on Control Technology and Applications (CCTA) | 978-1-6654-7338-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/CCTA49430.2022.9966119

model of the drone with respect to the non-inertial frame.

Observers for systems with unknown inputs have been studied for more than half a century. Several designs have been proposed for both linear and non-linear systems to jointly estimate the states and the unknown inputs [31]–[33]. In [34] the Kalman Filter is extended for the case of linear systems with unknown inputs. An Extended version of the Kalman Filter with Unknown Inputs (EKF-UI) is developed in [35]. The performance of the EKF-UI estimator was illustrated in several applications [36]–[38]. Since the drone dynamical model is in normal form, the Kalman Filter with Unknown Inputs (KF-UI) will be used in this work to estimate simultaneously the states of the drone and the moving environment accelerations.

Several Control strategies for quadrotors have been proposed in the literature. Linear control methods is widely used to control the drone by linearizing its dynamics around an operating point. In [39] PID and LQ are used while in [40] an LQR controller is proposed. In [41], [42], the authors consider the special case of controlling a drone that constantly accelerates, using a linear controller design that includes a triple integrator. Adaptive designs have also been proposed and tested in-flight, see [43]. [44] proposes a quaternion-based feedback control scheme based on a PD^2 feedback structure. While being easy to implement, linear controllers provide local convergence around the operating point only. On the other hand, non-linear controllers provide a wider flight range and better performance. Several non-linear controllers such as backstepping [45], sliding mode [46], and feedback linearization [47] demonstrate great performance in quadrotor control. In this work, we rely on the sliding mode controller due to its robustness properties and its capabilities to deal with disturbances and uncertainties. Sliding mode controller has been tested experimentally on a drone in presence of wind disturbances in [48] and has demonstrated good disturbance rejection. This choice is also motivated by the fact that no previous knowledge about the moving environment accelerations is needed when designing the control laws except their bounds.

The paper is organised as follows: First the problem is formulated in Section II. Then, the drone relative motion model in the non-inertial frame is presented in Section III. Section IV shows the design of the Kalman Filter with Unknown Input for estimating the states of the drone jointly with the non-inertial frame accelerations. On the other hand, Section V presents the sliding mode controller design for the drone in the moving environment. Two numerical simulations are presented in Section VI. Finally, concluding remarks and future work are given in Section VII.

II. Problem formulation

The aim of this paper is to make a drone track a trajectory inside a moving environment in a GPS-denied framework. The only measurements available are those obtained with respect to the non-inertial frame. The moving environment position, velocity, and acceleration are not assumed to be measured. In this paper we consider that the non-inertial frame **R'**(o',x',y',z') has only translation motion with respect to the inertial frame **R**(o,x,y,z). The simplest way to visualize this assumption is by considering that the drone is inside a box as illustrated in figure 1. The geostationary referential frame is considered in this work as the inertial referential frame **R**(o,x,y,z) and the non-inertial frame **R'**(o',x',y',z') is the box body frame.

Indeed, considering the moving environment as a box is enough to capture all the scenarios: If one wants to fly the drone in an elevator, the elevator is nothing else than a box that moves along the z-axis. If instead, the objective is to fly the drone inside a truck or a car moving on a straight road, this can be modeled by a drone inside a box that moves along x and y-axis.

Since the non-inertial frame is supposed to translate only, the linear positions and velocities of the drone are the only states affected. The angular positions and angular rates in the non-inertial frame are the same as the ones in the geostationary frame. In this work, we have access to the relative position of the drone with respect to the non-inertial frame as well as the angular position and velocities. During the experimental part of this activity, the angular positions and rates are measured using Pixhawk 4 instrumentation unit. [49]. The relative linear positions of the drone with respect to the moving environment are measured using onboard localization sensors. The linear velocities, on the other hand, are usually obtained by sensor fusion: the linear acceleration measurements are combined with position measurements to obtain the linear velocity of the drone. However, onboard accelerometers provide absolute linear acceleration, i.e. accelerations with respect to the inertial reference frame. Therefore, without any idea of the acceleration of the moving environment, one cannot deduce the relative acceleration of the drone in the non-inertial frame. Hence to implement a state feedback controller for trajectory tracking, an observer needs to be implemented to recover the unmeasured states.

Giving all the above, three problems need to be tackled:

1) *Modeling Problem:* modeling the drone dynamics in the non-inertial frame.
2) *Estimation Problem:* joint estimation of non-measurable states (linear velocities) and unknown inputs (non-inertial frame accelerations).
3) *Control Problem:* to make the drone follow a desired trajectory inside the moving environment.

The closed-loop system of the drone control inside the moving environment is illustrated by the block diagram in figure 2

III. Dynamical model

To derive the model of the drone in the non-inertial frame, we write three dynamical models that represent:

- The motion of the box with respect to the inertial frame **R**.
- The motion of the drone with respect to the inertial frame **R**.

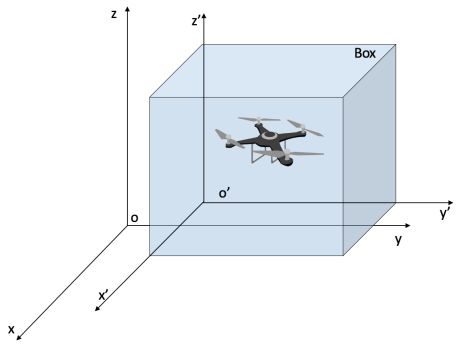

Fig. 1. The different frames of the problem

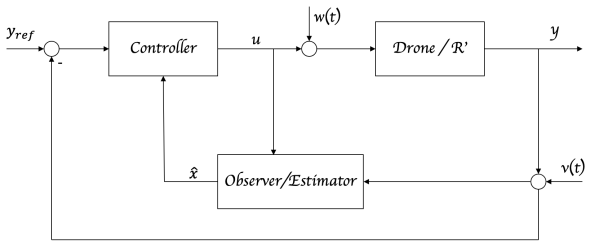

Fig. 2. Block diagram of the closed-loop control system of the drone in the moving environment.

- The motion of drone with respect to the non-inertial frame **R'**.

A. Equation of motion of the box w.r.t **R**

The box can be seen as a mass point and the equation of motion of the box w.r.t the inertial frame **R** can be written as:

$$m_b \frac{d\mathbf{v}_b}{dt} = \sum \mathbf{F}_{ext},$$

$$\begin{pmatrix} \ddot{x}_b \\ \ddot{y}_b \\ \ddot{z}_b \end{pmatrix} = \frac{1}{m_b} \begin{pmatrix} F_{bx} \\ F_{by} \\ F_{bz} \end{pmatrix}, \tag{1}$$

where F_{bx}, F_{by}, and F_{bz} are the external forces applied on the box along x, y and z-axis, respectively.
The state space representation of the box motion is given by:

$$\begin{pmatrix} \dot{x}_{b1} \\ \dot{x}_{b2} \\ \dot{x}_{b3} \\ \dot{x}_{b4} \\ \dot{x}_{b5} \\ \dot{x}_{b6} \end{pmatrix} = \begin{pmatrix} x_{b2} \\ \frac{1}{m_b} F_{bx} \\ x_{b4} \\ \frac{1}{m_b} F_{by} \\ x_{b6} \\ \frac{1}{m_b} F_{bz} \end{pmatrix}. \tag{2}$$

The state vector for the box motion in the inertial frame is given by
$(x_{b1}, x_{b2}, x_{b3}, x_{b4}, x_{b5}, x_{b6})^T = (x_b, \dot{x}_b, y_b, \dot{y}_b, z_b, \dot{z}_b)^T$.

B. Equation of motion of the drone w.r.t **R**

The equations of motion of the drone w.r.t the inertial frame **R** are given by [28]

$$\begin{aligned} \ddot{x}_d &= u_1(\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi) \\ \ddot{y}_d &= u_1(\cos\phi \sin\theta \sin\psi - \sin\psi \cos\psi) \\ \ddot{z} &= u_1(\cos\phi \cos\theta) - g \\ \ddot{\phi} &= a\dot{\theta}\dot{\psi} + u_3 \\ \ddot{\theta} &= b\dot{\psi}\dot{\phi} + u_4 \\ \ddot{\psi} &= c\dot{\theta}\dot{\phi} + u_5. \end{aligned} \tag{3}$$

where $a = \frac{I_y - I_z}{I_x}, b = \frac{I_z - I_x}{I_y}$ and $c = \frac{I_x - I_y}{I_z}$, and u_1, u_2, u_3, and u_4 are the inputs of the drone given by

$$\begin{aligned} u_1 &= \frac{b}{m}\left(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2\right) \\ u_2 &= \frac{b}{I_x}\left(\Omega_4^2 - \Omega_2^2\right) \\ u_3 &= \frac{b}{I_y}\left(\Omega_3^2 - \Omega_1^2\right) \\ u_4 &= \frac{l}{I_z}\left(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2\right) \end{aligned}.$$

ϕ, θ, and ψ represent respectively the pitch, roll, and yaw angles. Ω_i, i=1,2,3,4 represent the angular rates of the four rotors.
The state-space representation of the drone in the inertial frame is:

$$\begin{cases} \dot{x}_{d1} = x_{d2} \\ \dot{x}_{d2} = \left[\cos(x_7)\sin(x_9)\cos(x_{11}) + \sin(x_7)\sin(x_{11})\right] u_1 \\ \dot{x}_{d3} = x_{d4} \\ \dot{x}_{d4} = \left[\cos(x_7)\sin(x_9)\sin(x_{11}) - \sin(x_7)\cos(x_{11})\right] u_1 \\ \dot{x}_{d5} = x_{d6} \\ \dot{x}_{d6} = \left[\cos(x_7)\cos(x_9)\right] u_1 - g \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = a x_{10} x_{12} + u_2 \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = b x_8 x_{12} + u_3 \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = c x_8 x_{10} + u_4. \end{cases} \tag{4}$$

The state vector for the drone motion in the inertial frame is given by $(x_{d1}, x_{d2}, x_{d3}, x_{d4}, x_{d5}, x_{d6}, x_7, x_8, x_9, x_{10}, x_{11}, x_{12})^T = (x_d, \dot{x}_d, y_d, \dot{y}_d, z_d, \dot{z}_d, \phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi})^T$.

C. The drone dynamical model in the non-inertial frame **R'**

Since the box has exclusively translation motion with respect to the inertial frame, only the linear positions and velocities of the drone are affected. The angular positions and velocities remain unchanged with the change of frame.
The linear acceleration of the drone w.r.t the inertial frame **R** is expressed as the sum of the Box acceleration in **R** and the drone relative acceleration in **R'**

$$\begin{pmatrix} \ddot{x}_d \\ \ddot{y}_d \\ \ddot{z}_d \end{pmatrix} = \begin{pmatrix} \ddot{x}_b + \ddot{x}_r \\ \ddot{y}_b + \ddot{y}_r \\ \ddot{z}_b + \ddot{z}_r \end{pmatrix}. \tag{5}$$

Therefore the drone acceleration w.r.t R' can be expressed as:

$$\begin{pmatrix} \ddot{x}_r \\ \ddot{y}_r \\ \ddot{z}_r \end{pmatrix} = \begin{pmatrix} \ddot{x}_d - \ddot{x}_b \\ \ddot{y}_d - \ddot{y}_b \\ \ddot{z}_d - \ddot{z}_b \end{pmatrix}. \tag{6}$$

For the purpose of clear notation we consider the following notations for the rest of the paper:

$$\begin{pmatrix} \ddot{x}_b \\ \ddot{y}_b \\ \ddot{z}_b \end{pmatrix} = \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix}.$$

$$\mathrm{c}(x) = \cos(x),$$

$$\mathrm{s}(x) = \sin(x).$$

The state-space representation of the drone in the box frame is

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \left[\mathrm{c}\left(x_7\right)\mathrm{s}\left(x_9\right)\mathrm{c}\left(x_{11}\right) + \mathrm{s}\left(x_7\right)\mathrm{s}\left(x_{11}\right)\right]u_1 - a_x \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \left[\mathrm{c}\left(x_7\right)\mathrm{s}\left(x_9\right)\mathrm{s}\left(x_{11}\right) - \mathrm{s}\left(x_7\right)\mathrm{c}\left(x_{11}\right)\right]u_1 - a_y \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = \left[\mathrm{c}\left(x_7\right)\mathrm{c}\left(x_9\right)\right]u_1 - g - a_z \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = a x_{10} x_{12} + u_2 \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = b x_8 x_{12} + u_3 \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = c x_8 x_{10} + u_4, \end{cases}$$
$$y = \begin{pmatrix} x_1 & x_3 & x_5 & x_7 & x_8 & x_9 & x_{10} & x_{11} & x_{12} \end{pmatrix}^T. \tag{7}$$

$x = (x_r, \dot{x}_r, y_r, \dot{y}_r, z_r, \dot{z}_r, \phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi})^T$ is the state vector, and a_x , a_y, a_z represent the acceleration of the box with respect to x , y , and z axis, respectively.

System 13 has the following global structure:

$$\begin{cases} \dot{x} = Ax(t) + B(y)u(t) + f(y) + Ea(t) \\ y = Cx, \end{cases} \tag{8}$$

where $a(t) = [a_x, a_y, a_z]^T$ is the box acceleration. The matrices A, $B(y)$, E and C, and the vector field $f(y)$ are given by:

$$A = \begin{cases} a_{ij} = 1; & i = 1, 3, 5, 7, 9, 11 \\ & j = i + 1 \\ a_{ij} = 0; & else. \end{cases}$$

$$B(y) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \mathrm{c}\left(x_7\right)\mathrm{s}\left(x_9\right)\mathrm{c}\left(x_{11}\right) + \mathrm{s}\left(x_7\right)\mathrm{s}\left(x_{11}\right) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \mathrm{c}\left(x_7\right)\mathrm{s}\left(x_9\right)\mathrm{s}\left(x_{11}\right) - \mathrm{s}\left(x_7\right)\mathrm{c}\left(x_{11}\right) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \mathrm{c}\left(x_7\right)\mathrm{c}\left(x_9\right) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$E = \begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \\ \hline 0_{6\times 3} \end{pmatrix} \quad f(y) = \begin{pmatrix} 0_{6\times 1} \\ \hline 0 \\ a x_{10} x_{12} \\ 0 \\ b x_8 x_{12} \\ 0 \\ c x_8 x_{10} \end{pmatrix}$$

$$C = \left(\begin{array}{cccccc|c} 1 & 0 & 0 & 0 & 0 & 0 & \\ 0 & 0 & 1 & 0 & 0 & 0 & 0_{3\times 6} \\ 0 & 0 & 0 & 0 & 1 & 0 & \\ \hline \multicolumn{6}{c|}{0_6} & I_6 \end{array}\right)$$

IV. ESTIMATION PROBLEM

Considering the process and measurement noises, we rewrite system 8

$$\begin{cases} \dot{x} = Ax(t) + B(y)u(t) + f(y) + Ea(t) + w(t) \\ y = Cx + v(t), \end{cases} \tag{9}$$

where $w(t)$ is the process noise and $v(t)$ is the measurement noise.

Since the system is in normal form, we use Kalman Filter with Unknown Input (KF-UI) [35] to estimate the velocities of the drone in the non-inertial frame simultaneously with moving environment accelerations $a(t)$. Without loss of generality, we assume that $a(t)$ is piece-wise constant.

Considering the above, we augment the state space representation

$$\begin{cases} \dot{\xi} = \begin{pmatrix} A & E \\ 0_{3\times 12} & 0_{3\times 3} \end{pmatrix}\xi + \begin{pmatrix} B(y) \\ 0_{3\times 4} \end{pmatrix}u + \begin{pmatrix} f(y) \\ 0_{3\times 4} \end{pmatrix} \\ \qquad + \begin{pmatrix} w(t) \\ w_a(t) \end{pmatrix} \\ y = C_\xi \xi + v(t) \end{cases}, \tag{10}$$

where $\xi = [x \quad a]^T$ is the new augmented state vector, $w_a(t)$ is the acceleration's noise, and $C_\xi = [C \quad 0_{9\times 3}]$.

The augment system can be written under the compact state space representation:

$$\begin{cases} \dot{\xi} = A_\xi \xi + B_\xi(y)u(t) + f_\xi(y) + w_\xi(t) \\ y = C_\xi \xi + v(t), \end{cases} \tag{11}$$

The Kalman Filter with Unknown Input for system 11 is given by

$$\begin{cases} \dot{\hat{\xi}} = A_\xi \hat{\xi} + B_\xi(y)u(t) + f_\xi(y) + K(y - C_\xi \hat{\xi}) \\ K = PC^T R^{-1} \\ AP + PA^T - PC^T R^{-1} CP + Q = 0. \end{cases} \tag{12}$$

where R and Q represent covariance matrix of the measurement noise $v(t)$ and the covariance matrix of the process noise $w(t)$ and the unknown input noise $w_a(t)$, respectively.

V. CONTROL PROBLEM

To design a controller for the drone, we rewrite the state space representation as

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = u_x u_1 - a_x \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = u_y u_1 - a_y \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = [\mathrm{c}\,(x_7)\,\mathrm{c}\,(x_9)]\,u_1 - g - a_z \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = a x_{10} x_{12} + u_2 \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = b x_8 x_{12} + u_3 \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = c x_8 x_{10} + u_{4,} \end{cases} \tag{13}$$

where u_x and u_y are virtual control inputs defined as:

$$\begin{cases} u_x = \mathrm{c}\,(x_7)\,\mathrm{s}\,(x_9)\,\mathrm{c}\,(x_{11}) + \mathrm{s}\,(x_7)\,\mathrm{s}\,(x_{11}) \\ u_y = \mathrm{c}\,(x_7)\,\mathrm{s}\,(x_9)\,\mathrm{s}\,(x_{11}) - \mathrm{s}\,(x_7)\,\mathrm{c}\,(x_{11}) \end{cases} \tag{14}$$

Slotine and Li propose in [50] a general form of sliding surfaces given by

$$S(x) = \left(\frac{d}{dt} + \lambda\right)^{r-1} e(x),$$

where $e(x) = x - x_d$ is the tracking error, λ is a positive constant, and r is the relative degree of the system under consideration.

In accordance with the above recommendation, we consider the sliding surfaces

$$\begin{cases} S_x = \dot{e}_1 + \lambda_1 e_1 \\ S_y = \dot{e}_3 + \lambda_3 e_3 \\ S_z = \dot{e}_5 + \lambda_5 e_5 \\ S_\phi = \dot{e}_7 + \lambda_7 e_7 \\ S_\varphi = \dot{e}_9 + \lambda_9 e_9 \\ S_\psi = \dot{e}_{11} + \lambda_{11} e_{11} \end{cases}$$

where $e_i = x_i - x_{id}$, for $i = 1, 3, 5, 7, 9, 11$, are the tracking errors, and λ_i, $i = 1, 3, 5, 7, 9, 11$, are positive constants.

We consider the following assumption:
Assumption1
The non-inertial frame accelerations are bounded, i.e. $|a_x| \leq a_1$, $|a_y| \leq a_3$, $|a_z| \leq a_5$.

Control input for x :
Consider the following Lyapunov function:

$$V_1 = \frac{1}{2} S_x^T S_x,$$

$$\begin{gathered} \dot{V}_1 = S_x \dot{S}_x = S_x[\ddot{e}_1 + \lambda_1 \dot{e}_1], \\ \dot{V}_1 = S_x[\ddot{x}_1 - \ddot{x}_{1d} + \lambda_1\,(\dot{x}_1 - \dot{x}_{1d})], \\ \dot{V}_1 = S_x[\dot{x}_2 - \ddot{x}_{1d} + \lambda_1\,(x_2 - \dot{x}_{1d})], \\ \dot{V}_1 = S_x[u_x u_1 - a_x - \ddot{x}_{1d} + \lambda_1\,(x_2 - \dot{x}_{1d})], \end{gathered}$$

Since x_2 is not measured, we substitute it by its estimate $\hat{x}_2$

$$\begin{gathered} \dot{V}_1 = S_x\,[u_x u_1 - \ddot{x}_{1d} + \lambda_1\,(\hat{x}_2 - \dot{x}_{1d})] - S_x a_x, \\ \dot{V}_1 \leq S_x\,[u_x u_1 - \ddot{x}_{1d} + \lambda_1\,(\hat{x}_2 - \dot{x}_{1d})] + |a_x|\,|S_x|\,, \end{gathered}$$

Using **Assumption 1**

$$\dot{V}_1 \leq S_x\,(u_x u_1 - \ddot{x}_{1d} + \lambda_1\,(\hat{x}_2 - \dot{x}_{1d})) + a_1 |S_x|),$$

For $\dot{V}_1 \leq 0$ it's enough to choose :

$$\dot{V}_1 \leq S_x[\underbrace{u_x u_1 - \ddot{x}_{1d} + \lambda_1\,(\hat{x}_2 - \dot{x}_{1d}) + a_1\,\mathrm{sign}\,(S_x)}_{-k_1\,\mathrm{sign}(S_x)}],$$

$$u_x u_1 - \ddot{x}_{1d} + \lambda_1\,(\hat{x}_2 - \dot{x}_{1d}) + a_1\,\mathrm{sign}\,(S_x) = -k_1\,\mathrm{sign}\,(S_x)\,,$$

$$u_x = \frac{1}{u_1}\left[-\,(k_1 + a_1)\,\mathrm{sign}\,(S_x) + \ddot{x}_{1d} - \lambda_1\,(\hat{x}_2 - \dot{x}_{1d})\right].$$

Control Input for y and z:
Using the same steps as above, we find the control laws for y and z:

$$u_y = \frac{1}{u_1}\left[-\,(k_3 + a_3)\,\mathrm{sign}\,(S_y) + \ddot{x}_{3d} - \lambda_3\,(\hat{x}_4 - \dot{x}_{3d})\right].$$

$$\begin{aligned} u_1 = \frac{1}{\mathrm{c}\,(x_7)\,\mathrm{c}\,(x_9)}\,[&-(k_5 + a_5)\,\mathrm{sign}\,(S_z) + g \\ &+ \ddot{x}_{5d} - \lambda_5\,(\hat{x}_6 - \dot{x}_{5d})]. \end{aligned}$$

For x_7 and $x_9 \neq \frac{\pi}{2}$

Control input for ϕ :

$$V_7 = \frac{1}{2} S_\phi^T S_\phi,$$

$$\dot{V}_7 = S_\phi[\ddot{e}_7 + \lambda_7 \dot{e}_7] = S_\phi[\dot{x}_8 - \ddot{x}_{7d} + \lambda_7\,(x_8 - \dot{x}_{7d})],$$

$$\dot{V}_7 = S_\phi[a x_{10} x_{12} + u_2 - \ddot{x}_{7d} + \lambda_7\,(x_8 - \dot{x}_{7d})],$$

For $\dot{V}_7 \leq 0$ it is enough to choose :

$$a x_{10} x_{12} + u_2 - \ddot{x}_{7d} + \lambda_7\,(x_8 - \dot{x}_{7d}) = -k_7\,\mathrm{sign}(\mathrm{S}_\phi),$$

$$u_2 = -k_7\,\mathrm{sign}\,(S_\phi) - a x_{10} x_{12} + \ddot{x}_{7d} - \lambda_7\,(x_8 - \dot{x}_{7d})\,.$$

Control input for θ and ψ:
Using the same steps as above:

$$u_3 = -k_9\,\mathrm{sign}\,(S_\theta) - b x_8 x_{12} + \ddot{x}_{9d} - \lambda_9\,(x_{10} - \dot{x}_{9d})\,.$$

$$u4 = -k_{11}\,\mathrm{sign}\,(S_\psi) - c x_8 x_{10} + \ddot{x}_{11d} - \lambda_{11}\,(x_{12} - \dot{x}_{11d})\,.$$

Finally, the control laws for the drone trajectory tracking inside the box are summarized below

$$\begin{cases} u_1 = \frac{1}{\mathrm{c}(x_7)\mathrm{c}(x_9)}[-(k_5 + a_5)\,\mathrm{sign}\,(S_z) + g + \ddot{x}_{5d} \\ \qquad\qquad -\lambda_5\,(\hat{x}_6 - \dot{x}_{5d})] \\ u_2 = -k_7\,\mathrm{sign}\,(S_\phi) - a x_{10} x_{12} + \ddot{x}_{7d} - \lambda_7\,(x_8 - \dot{x}_{7d}) \\ u_3 = -k_9\,\mathrm{sign}\,(S_\theta) - b x_8 x_{12} + \ddot{x}_{9d} - \lambda_9\,(x_{10} - \dot{x}_{9d}) \\ u_4 = -k_{11}\,\mathrm{sign}\,(S_\psi) - c x_8 x_{10} + \ddot{x}_{11d} \\ \qquad -\lambda_{11}\,(x_{12} - \dot{x}_{11d}) \\ u_x = \frac{1}{u_1}\left[-\,(k_1 + a_1)\,\mathrm{sign}\,(S_x) + \ddot{x}_{1d} - \lambda_1\,(\hat{x}_2 - \dot{x}_{1d})\right] \\ u_y = \frac{1}{u_1}\left[-\,(k_3 + a_3)\,\mathrm{sign}\,(S_y) + \ddot{x}_{3d} - \lambda_3\,(\hat{x}_4 - \dot{x}_{3d})\right]. \end{cases} \tag{15}$$

The desired trajectories x_{1d}, x_{3d}, x_{5d}, and x_{11d} are set a-priori. However, x_{7d} and x_{9d} are obtained from the virtual control laws u_x and u_y as follows:

$$\begin{bmatrix} s\left(x_{7d}\right) \\ c\left(x_{7d}\right) s\left(x_{9d}\right) \end{bmatrix} = \begin{bmatrix} s\left(x_{11d}\right) & -c\left(x_{11d}\right) \\ c\left(x_{11d}\right) & s\left(x_{11d}\right) \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix}.$$

VI. Numerical Simulation

In this section two simulations are conducted. The first one considers the accelerations of the box to be constant (ideal case). The second simulation deals with a more concrete example, i.e. flying the drone in an elevator. The drone parameters used for the simulation are given in table I. Additionally, in the implementation of the siding mode control laws sign(S) was substituted by tanh(S/0.1).

TABLE I
SIMULATION PARAMETERS [51]

Name	Symbol	Value	Unit
Mass	m	1	Kg
Distance between the rotor and the center of mass	l	0.24	m
Moment of inertia along x	I_x	8.1×10^{-3}	Kg.m^2
Moment of inertia along y	I_y	8.1×10^{-3}	Kg.m^2
Moment of inertia along z	I_z	14.2×10^{-3}	Kg.m^2

A. Ideal case: constant accelerations

In this simulation, the box has constant accelerations in the x, y, and z directions. The simulation was conducted considering the following:

- Sampling time Ts=0.001s
- Box acceleration a_x=1 m/s^2, a_y=2 m/s^2 and a_z=3 m/s^2.
- Measurement noise covariance Matrix R=1,
- Process and unknown input noise covariance matrix $Q = \begin{pmatrix} 5I_{12} & 0_{12\times3} \\ 0_{3\times12} & 50I_3 \end{pmatrix}$
- Gaussian measurement noise $v(t)$, with a mean of 0.01
- Gaussian Process noise $w(t)$, with a mean of 0.001
- Gaussian noise for the box acceleration $w_a(t)$, with a mean of 0.0001
- ξ_0 =[0.5$\times$ ones(1,12) a_x a_y a_z]^T
- $\hat{\xi}_0$=$0_{15\times1}$

As shown in figure 3, the estimated linear velocities using the KF-UI converge to their real values in approximately 5s. on the other hand, the combined KF-UI and Sliding Mode controller force the closed-loop system to converge to its desired position in about 4s.

B. Elevator Example

The elevator acceleration was measured using Pixhawk 4 flight controller's internal Inertial Measurement Unit (IMU) with a sampling time of 0.02s. The Pixhawk was connected to the computer using a serial port to collect the data, and recorded with the Robot Operating System (ROS) recording tools. To obtain the real elevator acceleration, 9.81 was

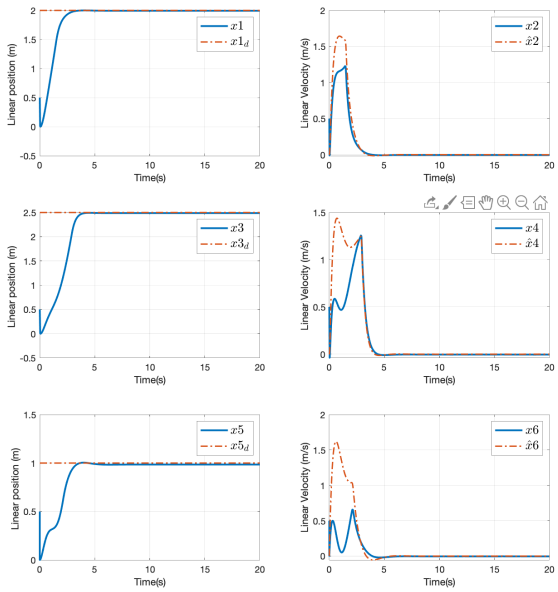

Fig. 3. Tracking performance of the linear positions and estimation of the linear velocities inside the box with constant accelerations using the observer-based sliding mode controller

deducted from the measurements. The measurement collection experiment was conducted on one of the elevators of King Abdullah University of Science and Technology campus for 593 s. The measured accelerations are then used in the the foregoing simulation environment. The process and measurement noises remain the same as in the previous simulation. Figure 4 shows the altitude trajectory tracking in the elevator along with the elevator's acceleration and its reconstructed value using KF-UI. Figure 5 illustrates the tracking performance of the combined KF-UI and sliding mode controller along x' and y'-axis. The estimated linear velocities in the non-inertial frame as well as their estimation errors are given in figure 6.

From figure 4 and figure 5, one can see that the combined KF-UI and Sliding Mode controller force the linear positions of the drone to converge to their desired values, despite the small estimation errors of the linear velocities (figure 6). This demonstrates the robustness of the sliding mode controller against estimation errors. Moreover, figure 4 gives a better illustration of the Sliding Mode controller performance and robustness, where it was able to react to the big acceleration spikes without considerably affecting the altitude of the drone inside the elevator.

VII. Conclusion

In this paper, we proposed a formulation for the problem of flying drones in a moving environment that has translational motion with respect to the geostationary reference frame. The moving environment is associated with a non-inertial frame. Based on the relative motion principles, a model for the drone motion with respect to the non-inertial frame was

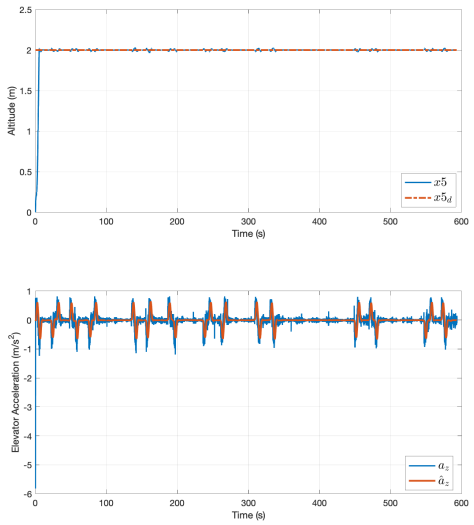

Fig. 4. Altitude tracking in the moving elevator using the combined KF-UI and Sliding Mode controller.

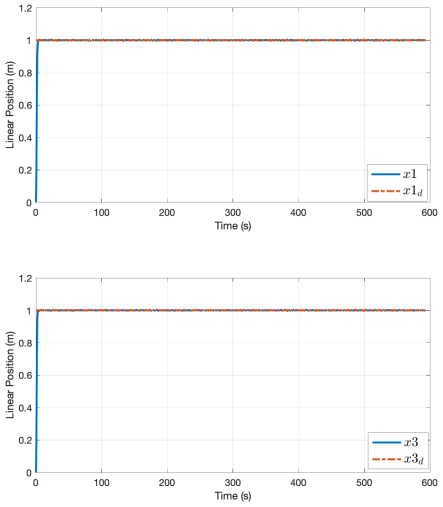

Fig. 5. Position tracking along x and y-axis in the elevator using the combined KF-UI and Sliding Mode controller.

developed. A Kalman Filter with Unknown Inputs (KF-UI) was used to estimate the linear velocities of the drone as well as the moving environment accelerations. The Estimated velocities were then fed to a sliding mode controller along with the other measured states to make the drone track the desired trajectory. The combined KF-UI and Sliding Mode control demonstrated great performance in making the quadrotor's linear positions converge to the desired value despite the classical chattering drawback of the sliding mode controller. As future work, we intend to extend the relative motion model of the drone to the more general case of a moving environment that has both translation and rotational motions in the inertial reference frame. Additionally, further

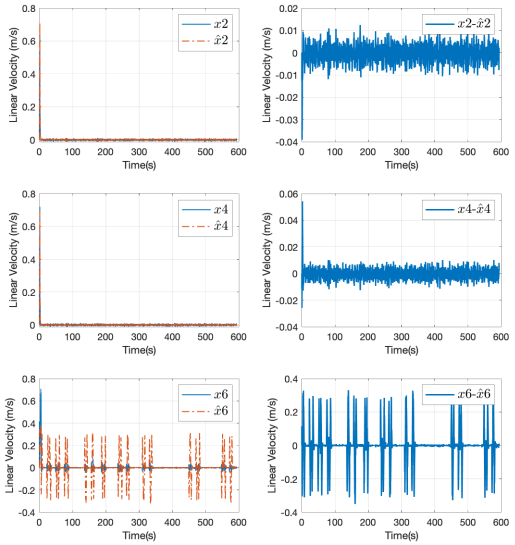

Fig. 6. Linear velocities estimation using the KF-UI and their estimation errors.

experiments to validate the observer-based control strategy used are planned for the near future.

ACKNOWLEDGMENT

Research reported in this publication was supported by King Abdullah University of Science and Technology (KAUST) with the Base Research Fund (BAS/1/1627-01-01, BAS/1/1682-01-01), and KAUST AI-Initiative. The authors would also like to thank Mr. Olivier Toupet from the Jet Propulsion Laboratory for suggesting the idea for this research.

REFERENCES

[1] Steven Rasmussen, Krishnamoorthy Kalyanam, Satyanarayana Manyam, David Casbeer, and Christopher Olsen. Practical considerations for implementing an autonomous, persistent, intelligence, surveillance, and reconnaissance system. In *2017 IEEE Conference on Control Technology and Applications (CCTA)*, pages 1847–1854, 2017.

[2] Joshua Shaffer, Estefany Carrillo, and Huan Xu. Receding horizon synthesis and dynamic allocation of UAVs to fight fires. In *2018 IEEE Conference on Control Technology and Applications (CCTA)*, pages 138–145, 2018.

[3] Eric Cheng. *Aerial photography and videography using drones*. Peachpit Press, 2015.

[4] Daniel Câmara. Cavalry to the rescue: Drones fleet to help rescuers operations over disasters scenarios. In *2014 IEEE Conference on Antenna Measurements & Applications (CAMA)*, pages 1–4. IEEE, 2014.

[5] Yunus Karaca, Mustafa Cicek, Ozgur Tatli, Aynur Sahin, Sinan Pasli, Muhammed Fatih Beser, and Suleyman Turedi. The potential use of unmanned aircraft systems (drones) in mountain search and rescue operations. *The American journal of emergency medicine*, 36(4):583–588, 2018.

[6] Pere Molina, Ismael Colomina, Pedro Victoria, Jan Skaloud, Wolfgang Kornus, Rafael Prades, and Carmen Aguilera. Drones to the rescue! Technical report, 2012.

[7] UM Rao Mogili and BBVL Deepak. Review on application of drone systems in precision agriculture. *Procedia computer science*, 133:502–509, 2018.

[8] Pratap Tokekar, Joshua Vander Hook, David Mulla, and Volkan Isler. Sensor planning for a symbiotic UAV and UGV system for precision agriculture. *IEEE Transactions on Robotics*, 32(6):1498–1511, 2016.

[9] Shreeram Marathe. Leveraging drone based imaging technology for pipeline and RoU monitoring survey. In *SPE Symposium: Asia Pacific Health, Safety, Security, Environment and Social Responsibility*. OnePetro, 2019.

[10] Oswaldo Menéndez, Marcelo Pérez, and Fernando Auat Cheein. Visual-based positioning of aerial maintenance platforms on overhead transmission lines. *Applied Sciences*, 9(1):165, 2019.

[11] Jarrod C Hodgson, Rowan Mott, Shane M Baylis, Trung T Pham, Simon Wotherspoon, Adam D Kilpatrick, Ramesh Raja Segaran, Ian Reid, Aleks Terauds, and Lian Pin Koh. Drones count wildlife more accurately and precisely than humans. *Methods in Ecology and Evolution*, 9(5):1160–1167, 2018.

[12] Julie Linchant, Jonathan Lisein, Jean Semeki, Philippe Lejeune, and Cédric Vermeulen. Are unmanned aircraft systems (UAS s) the future of wildlife monitoring? a review of accomplishments and challenges. *Mammal Review*, 45(4):239–252, 2015.

[13] Qiuhong Wang, Jingjing Gu, Haitao Huang, and Yanchao Zhao. Online drone-based moving target detection system in dense-obstructer environment. In *2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)*, pages 860–867. IEEE, 2018.

[14] Tyler Wall and Torin Monahan. Surveillance and violence from afar: The politics of drones and liminal security-scapes. *Theoretical criminology*, 15(3):239–254, 2011.

[15] Gregory S McNeal. Drones and the future of aerial surveillance. *Geo. Wash. L. Rev.*, 84:354, 2016.

[16] Milan Erdelj, Enrico Natalizio, Kaushik R Chowdhury, and Ian F Akyildiz. Help from the sky: Leveraging UAVs for disaster management. *IEEE Pervasive Computing*, 16(1):24–32, 2017.

[17] Geoffrey Ling and Nicole Draghic. Aerial drones for blood delivery. *Transfusion*, 59(S2):1608–1611, 2019.

[18] Evan Ackerman and Eliza Strickland. Medical delivery drones take flight in East Africa. *IEEE Spectrum*, 55(1):34–35, 2018.

[19] The Action Lab. What Happens If You Fly a Drone In An Elevator? Real Experiment!, Youtube, 2019. [Video file]. Available: `https://www.youtube.com/watch?v=DUGwdcgi2L8` [Accessed 16-Feb-2021].

[20] Ken Heron. Hovering Drone in a Van - Will it move with it? - KEN HERON, Youtube, 2018. [Video file]. Available: `https://www.youtube.com/watch?v=LEgJBDxr3uU` [Accessed 16-Feb-2021].

[21] The Action Lab. If You Fly a Drone in a Car, Does it Move With It? (Dangerous In-Car Flight Challenge), Youtube, 2017. [Video file]. Available: `https://www.youtube.com/watch?v=XjTj-tGPSWE` [Accessed 16-Feb-2021].

[22] Sinatra314. Flying a drone in an elevator. What happens?, Youtube, 2020. [Video file]. Available: `https://www.youtube.com/watch?v=kvazr0W8-Bc` [Accessed 16-Feb-2021].

[23] DroningON. DroningON — Drone In An Elevator/Lift - Ryze/DJI Tello Experiment, Youtube, 2018. [Video file]. Available: `https://www.youtube.com/watch?v=vbgHPV4G0Sc` [Accessed 16-Feb-2021].

[24] Qi Lu, Beibei Ren, and Siva Parameswaran. Shipboard landing control enabled by an uncertainty and disturbance estimator. *Journal of Guidance, Control, and Dynamics*, 41(7):1502–1520, 2018.

[25] Marco Tognon, Sanket S. Dash, and Antonio Franchi. Observer-based control of position and tension for an aerial robot tethered to a moving platform. *IEEE Robotics and Automation Letters*, 1(2):732–737, 2016.

[26] Fady Alami, Abdulrahman Hussian, and Naim Ajlouni. Design and implementation of a multi-stage PID controller for non-inertial referenced UAV. *Electrica*, 20(2):199–207, 2020.

[27] Hakim Bouadi, M. Bouchoucha, and M. Tadjine. Modelling and stabilizing control laws design based on backstepping for an uav type-quadrotor. *IFAC Proceedings Volumes*, 40:245–250, 12 2007.

[28] Samir Bouabdallah and Roland Siegwart. Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. volume 2005, pages 2247–2252, 01 2005.

[29] Abdelhamid Chriette. Contribution à la commande et à la modélisation des hélicoptères : asservissement visuel et commande adaptative. 2001.

[30] Matko Orsag and Stjepan Bogdan. Hybrid control of quadrotor. In *2009 17th Mediterranean Conference on Control and Automation*, pages 1239–1244, 2009.

[31] Ron J. Patton, Paul M. Frank, and Robert N. Clarke. *Fault Diagnosis in Dynamic Systems: Theory and Application*. Prentice-Hall, Inc., USA, 1989.

[32] Ron Patton, P. Frank, and R. Clark. *Issue of Fault Diagnosis for Dynamic Systems*. 01 2000.

[33] A.M. Pertew, H.J. Marquez, and Q. Zhao. Design of unknown input observers for Lipschitz nonlinear systems. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 4198–4203 vol. 6, 2005.

[34] Shuwen Pan. *System identification and damage detection of structures with unknown excitations*. PhD thesis, 2006.

[35] Daniel Simon. Optimal state estimation: Kalman, H-infinity, and nonlinear approaches. 01 2006.

[36] Sarah Mechhoud, Emmanuel Witrant, Luc Dugard, and Didier Moreau. Estimation of heat source term and thermal diffusion in Tokamak plasmas using a Kalman filtering method in the early lumping approach. *IEEE Transactions on Control Systems Technology*, 23(2):449–463, 2015.

[37] Junn Loo, Ze Yang Ding, Evan Davies, Surya Nurzaman, and Chee Pin (Edwin) Tan. Curvature and force estimation for a soft finger using an EKF with Unknown Input Optimization. 07 2020.

[38] Zehor Belkhatir, S. Mechhoud, and Taous-Meriem Laleg-Kirati. Kalman filter based estimation algorithm for the characterization of the spatiotemporal hemodynamic response in the brain. *Control Engineering Practice*, 89:180–189, 08 2019.

[39] S. Bouabdallah, A. Noth, and R. Siegwart. PID vs LQ control techniques applied to an indoor micro quadrotor. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, volume 3, pages 2451–2456 vol.3, 2004.

[40] Ian D. Cowling, Oleg A. Yakimenko, James F. Whidborne, and Alastair K. Cooke. A prototype of an autonomous controller for a quadrotor UAV. In *2007 European Control Conference (ECC)*, pages 4001–4008, 2007.

[41] Juan-Pablo Afman, Eric Feron, and John Hauser. Nonlinear maneuver regulation for reduced-G atmospheric flight. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 731–736, 2018.

[42] Juan-Pablo Afman, Eric Feron, and John Hauser. Triple-integral control for reduced-G atmospheric flight. In *2018 Annual American Control Conference (ACC)*, pages 392–397, 2018.

[43] Siddhardha Kedarisetty and Joel George Manathara. Acceleration control of a multi-rotor UAV towards achieving microgravity. *Aerospace Systems*, 2(2):175–188, 2019.

[44] Abdelhamid Tayebi and S. McGilvray. Attitude stabilization of a four-rotor aerial robot. volume 2, pages 1216 – 1221 Vol.2, 01 2005.

[45] Samir Bouabdallah and Roland Siegwart. Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. volume 2005, pages 2247–2252, 01 2005.

[46] T. Madani and A. Benallegue. Backstepping sliding mode control applied to a miniature quadrotor flying robot. In *IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics*, pages 700–705, 2006.

[47] Saif A. Al-Hiddabi. Quadrotor control using feedback linearization with dynamic extension. In *2009 6th International Symposium on Mechatronics and its Applications*, pages 1–3, 2009.

[48] Tao Jiang, Tao Song, and Defu Lin. Integral sliding mode based control for quadrotors with disturbances: Simulations and experiments. *International Journal of Control, Automation and Systems*, 17, 05 2019.

[49] PX4 autopilot. Px4 system architecture, March 2021. Available: `https://docs.px4.io/master/en/concept/px4_systems_architecture.html` [Accessed 18-Feb-2022].

[50] Jean-Jacques E Slotine and Weiping Li. *Applied nonlinear control*. 1991.

[51] M. Belkheiri, A. Rabhi, A. El Hajjaji, and C. Pegard. Different linearization control techniques for a quadrotor system. In *CCCA12*, pages 1–6, 2012.