Design, implementation, and evaluation of position controllers for multirotor vehicles

Maria Leonor Albergaria Rodrigues Morais Carvalho
leonor.carvalho@tecnico.ulisboa.pt
Instituto Superior Técnico, Universidade de Lisboa
November 2023

***Abstract*—The quadrotor is an UAV with numerous everyday applications whose control system has become a popular field of study. The quadrotor is an underactuated system with nonlinear dynamics and is typically controlled using an inner and outer loop structure. There are various techniques used to control the vehicle but, recently, the state of the art of trajectory tracking does not include a complete comparison of the different position and orientation controllers.**

This thesis designs, implements, and evaluates trajectory tracking controllers for the quadrotor, subject to the same environmental constraints, ensuring a fair comparison. Firstly, variations based on Backstepping (BS) and Sliding Mode (SM) of a controller with an inner and outer loop are developed, using Lyapunov's theory to prove stability. The controllers are implemented using Simulink and PX4 Autopilot; using Gazebo to create realistic simulation environments, the controllers are evaluated when following trajectories, some of which are under the influence of the wind. Additionally, an adaptive controller was proposed, considering only the outer loop of the system, in order to evaluate the error response when the mass parameter suffers a modification. The metrics used for the evaluation are the mean and integral absolute errors, and the settling time of the position error.

The conclusions drawn confirm that, of all the controllers, the Proportional Integral Derivative (PID) variation, and of all the non-integral controllers, the SM hold the best performances in trajectory tracking and rejection of disturbances.

***Index Terms*—Quadrotor, trajectory tracking, nonlinear, inner-outer loop**

I. Introduction

Multirotor vehicles are a popular field of study in robotics and, as technology advances over time, the research on how to improve the control of these vehicles grows continuously.

Multirotor vehicles possess more than two rotors, which generate torque and thrust that allow the vehicle to move. Among these robots, there exists a particular interest in the quadrotor (four-rotor) UAV [1] and, because of its hovering capability, the multirotor aircraft is widely used in various environments where it is important to guarantee a stationary flight, such as traffic monitoring, aerial photography [2], [3], construction management [4] among others. The numerous applications of multirotors justify the objective of improving the trajectory tracking controller of these vehicles. However, this is a challenge. The quadrotor is an underactuated system with nonlinear dynamics and depending on the task given to the vehicle, the appropriate controller design may vary. There exist several techniques proven to control the movement of quadrotors, even though they have varying degrees of stability, tolerance to disturbances, performance, etc. Nevertheless, it becomes almost impossible to compare the different strategies, mainly because the experiments and simulations demonstrated in the literature are not subject to the same conditions, the characteristics of each quadrotor may be distinct, the experimental tests focus on different capabilities of the vehicle (maintaining hover state or following a trajectory) and the controller performance can be evaluated through different methods: error comparison, overshoot analysis, rise time, response to different functions and so on. This work identifies the trajectory tracking controllers commonly adopted for the aircraft. The main objective is to carry out an evaluation of the controllers and determine the strengths and weaknesses of each so that it is possible to experimentally test and validate the performance of all the presented controllers. First, the controllers are designed and tested in a realistic simulation, using the SITL (software-in-the-loop) version of the PX4 Autopilot and Gazebo, with the controllers implemented in Simulink. The PX4 SITL ensures that the code of the implemented controllers will run in any vehicle equipped with the PX4 Autopilot hardware. For the simulation environments, the Gazebo simulator was used to create realistic scenarios, including the quadrotor model and its sensors, so that we can be confident that the algorithms implemented will also run on real systems. The metrics used in the trajectory tracking evaluation are the Mean Absolute Error (MAE), Integral Absolute Error (IAE), and the value of the Settling Time (ST) of the position error.

II. Quadrotor Dynamic Model

This section presents the theory behind the dynamic model of the quadrotor. The basic dynamic model of the vehicle is derived assuming the quadrotor is a symmetrical rigid body and the time dependence is omitted, to simplify the notation. First, it is necessary to define the reference frames used to describe the model: the inertial frame, $\{I\}$, which is a fixed referential; the body frame, $\{B\}$, which is assumed to be attached to the quadrotor and, therefore, its displacement coincides with that of the robot. The inertial position of the multirotor is given by the vector $p = [xyz]^T$ and the Euler angles, roll, pitch and yaw, $\lambda = [\phi\, \theta\, \psi]^T$, describe the orientation of the quadrotor, also in the inertial frame. The quadrotor consists of four rotors, two of which rotate clockwise and the other two counterclockwise. Figure 1 illustrates the studied model.

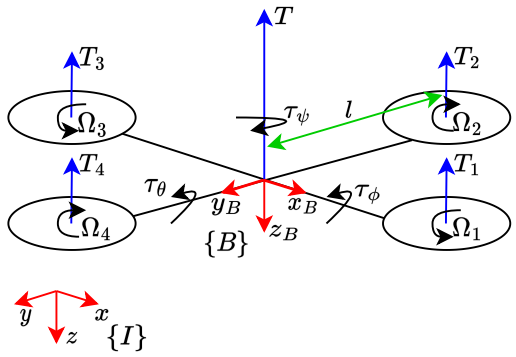


Fig. 1. Quadrotor model, mapping of the thrusts, torques and angular velocities of the rotors which characterise the vehicle.

Each rotor, as it rotates, produces thrust and reaction torque, T_n and Q_n,

$$T_n = c_T \Omega_n^2, \tag{1}$$

$$Q_n = c_Q \Omega_n^2, \tag{2}$$

that are directly proportional to the squared angular speed of the propellers, Ω_n. The constants c_T and c_Q can be discovered from static thrust experiments, that way including the rotor drag effect [5].

The moment $\tau_c = [\tau_\phi \ \ \tau_\theta \ \ \tau_\psi]^T$, describes the angular acceleration about the x_B, y_B, and z_B axis in the body frame, created by varying the different thrusts. The control variables of the model, $[T \ \tau_\phi \ \ \tau_\theta \ \ \tau_\psi]^T$, have the following relation,

$$\begin{bmatrix} T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & \ell & 0 & -\ell \\ \ell & 0 & -\ell & 0 \\ c & -c & c & -c \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix}, \tag{3}$$

in which ℓ refers to the distance between a rotor and the centre of the body and c depends on thrust and reaction torque constants, related to the drag effect. The linear relation between the actuation variables and the thrusts, T_i, influenced the selection of the actuation $[T \, \tau_\phi \, \tau_\theta \, \tau_\psi]^T$. Note that the total force, as it is always aligned with the z body axis, consists of the thrust, T , which is a scalar.

A. Kinematics of the Model

The kinematics are described by the rigid body linear

$$\dot{p} = Rv, \tag{4}$$

and angular

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

equations of motion. In (4) and (5), $v \in \mathbb{R}^3$ is the vector of linear velocities and $\omega \in \mathbb{R}^3$ is the vector of angular velocities, both expressed in the body frame. The matrix $S(\cdot)$ is skew-symmetric and simplifies calculations by transforming cross-products into the following matrix operation,

$$a \times b = S(a)b = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} b, \tag{6}$$

for all a and $b \in \mathbb{R}^3$.

The matrix

$$R = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - c\phi s\psi & s\psi s\phi + c\psi c\phi s\theta \\ c\theta s\psi & c\psi c\phi + s\psi s\theta s\phi & c\phi s\psi s\theta - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \tag{7}$$

corresponds to the rotation matrix from $\{B\}$ to $\{I\}$. Finally, addressing the orientation angles vector, λ, it is possible to relate its derivative with ω,

$$\dot{\lambda} = Q\omega, \tag{8}$$

with Q consisting of the transition matrix

$$Q = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & \frac{s\phi}{c\theta} & \frac{c\theta}{c\phi} \end{bmatrix}. \tag{9}$$

The notation $c\cdot$, $s\cdot$ and $t\cdot$ refers to the cosine, sine and tangent functions, respectively. The controllers which make use of the Q matrix or the Euler angles must avoid $\phi = \pm\frac{\pi}{2}$ or $\theta = \pm\frac{\pi}{2}$ because, in this case, Q is not defined. In the case of typical manoeuvres, such as those considered in this thesis, we are always a long way from this situation, and it is therefore not an obstacle to using these controllers.

B. Dynamics of the Model

The total force applied to the quadrotor, F, expressed in $\{B\}$, takes into account the gravitational force, the total thrust produced by the rotating propellers, T, and environment disturbances h, defined in $\{I\}$. Thus,

$$F = mgR^T e_3 - Te_3 + R^T h, \tag{10}$$

in which m, g and e_3 correspond to the mass of the vehicle, gravitational constant and the unit vector in the direction of the z axis, respectively. When applying the Second Law of Newton, the translational dynamics become

$$\dot{v} = -S(\omega)v + gR^T e_3 - \frac{T}{m}e_3 + R^T\frac{h}{m}, \tag{11}$$

or, in the inertial frame,

$$\ddot{p} = ge_3 - \frac{T}{m}r_3 + \frac{h}{m}, \tag{12}$$

with $Re_3 = r_3$.

Torques are responsible for the rotational motion of the body. For the quadrotor dynamic model it is important to distinguish between the control torque, $\tau_c = [\tau_\phi \ \ \tau_\theta \ \ \tau_\psi]^T$, and other moments induced by the dynamics of the model, τ, for example, the gyroscopic torque caused by the change in orientation of the propellers [6]. However, in this work, τ is neglected. Then, the total torque acting on the body is

$$n = \tau_c. \tag{13}$$

The rotational dynamics are described as

$$J\dot{\omega} = -S(\omega)J\omega + n, \tag{14}$$

with the diagonal matrix of the moments of inertia of the body denoted by

$$J = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix}. \tag{15}$$

The matrix J may be considered diagonal without compromising generality, because there is always an orientation of the referential $\{B\}$ in which J is diagonal.

III. Sate of the Art

There is a wide range of proposed and widely used controllers based on Proportional Integral Derivative (PID), Linear Quadratic Regulator (LQR), Backstepping (BS), Sliding Mode (SM), Model Predictive Control (MPC), adaptive, and more techniques, which may result in a good performance in the attitude stabilisation and position tracking of the vehicle.

A. Proportional Integral Derivative, PID

The PID controller is a type of linear controller which presents a simple structure, following the control law

$$\hat{u}(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt} + u_0, \tag{16}$$

where K_p, K_i, $K_d > 0$ and u_0 correspond to the proportional, integral, derivative gains, and the operating point around which the PID works, respectively. The value of u_0 compensates the gravitational force and the variable $e(t)$ represents the error. The model of the vehicle is represented by equations (12) and (14). For this explanation it is assumed that the disturbances h are null, and (12) is simplified into

$$\ddot{p} = u + ge_3, \tag{17}$$

where $u = \frac{-T}{m} r_3$. Assuming that a sufficiently fast inner-loop attitude controller is implemented, such that, from the point of view of the linear dynamics, $r_3 \approx r_{3d}$, one can define a control law for u that dictates the necessary thrust, T, and orientation, r_{3d} for the rotors to maintain the desired state. The reference to be tracked consists on the desired position, p_d, and the objective is to define a control law, u, to track the given position. Then, the error system corresponds to,

$$e = p - p_d \tag{18}$$

$$\dot{e} = \dot{p} - \dot{p}_d \tag{19}$$

$$\ddot{e} = u + ge_3 - \ddot{p}_d. \tag{20}$$

If u is established as

$$u = k_P e + k_D \dot{e} + k_I \int e - g + \ddot{p}_d, \tag{21}$$

the closed-loop error system, in time domain, becomes

$$\ddot{e} = k_P e + k_D \dot{e} + k_I \int e, \tag{22}$$

or

$$\ddot{p} = k_P e + k_D \dot{e} + k_I \int e + \ddot{p}_d, \tag{23}$$

in which $\ddot{p}_d$ is a feedforward term.

The dynamic system could also be controlled using a PD controller. However, there may be a constant error or time-varying error depending on the disturbances. The existence of the integral action allows the rejection of constant disturbances. The integral gain may have its downsides also, such as the appearance of overshoot, which is not a desired feature. Although the position control is important for the tracking problem, the attitude control is crucial so that the quadrotor tracks the desired trajectory. In Linear Control, the dynamic model is linearized about the hover condition.

Evidently there exist several projects ([7], [8], and [9]) which tackle the trajectory tracking problem using the PID technique. However, the quadrotor model is linearized around a desired condition, being the hover state the most exploited. Therefore, the PID controller can provide stability if the end goal is for the quadrotor to fly small-angle paths that need little manoeuvrability. In contrast, for tasks that require high manoeuvrability, such as acrobatic flights, the PID procedure may not provide the best solution because, in this situation, the model is far from the operating point and the linearization does not approximate well the nonlinear system.

B. Backstepping

Backstepping is a recursive technique widely used in nonlinear control design. For this method, the system is, generally, required to be in the strict-feedback form [10]. The quadrotor is a system that can be written in this form, and therefore the theory of backstepping can be applied to control the vehicle. The backstepping method is a fairly popular solution for the quadrotor position and attitude tracking and continues to evolve, so that the controller becomes more robust. Conventional backstepping, [11], and [12], may offer promising results, in the sense that the states converge asymptotically and provides fast responses. However, the model parameters may be difficult to choose and the robustness to uncertainties and disturbances could be improved. This method can be enhanced by adding integral action, resulting in integral backstepping [13], [14]. In addition to the advantages of the conventional backstepping, integral backstepping provides good disturbance rejection and allows the steady-state errors to disappear. Finally, there exist adaptive BS controllers [15], [16], and [17], that can estimate variations of the model parameters, which is extremely important for the current model.

C. Sliding Mode Controller, SMC

Sliding Mode control is a technique widely used in nonlinear systems. According to [18], the sliding mode controls the system trajectories by forcing and constraining them onto a predefined surface or manifold. Firstly, it is necessary to establish a sliding surface. This surface is defined by

$$s = 0, \tag{24}$$

and while the system states are on the surface, the system has the desired behaviour. Considering the quadrotor trajectory tracking problem, the sliding surface is designed according to the state errors of the system and depending on the desired subsystem to control, these could be position and/or attitude errors. Then, the control law is elaborated. For instance, let s be a sliding surface defined by the position, e_p, and linear velocity, $e_v = \dot{e}_p$, errors, resulting in

$$s = \lambda e_p + e_v, \tag{25}$$

with λ being a positive gain. The idealised behaviour of the system in SMC presupposes that the system trajectories stay constricted to the surface, $s = 0$, even if there exists uncertainties in the dynamic model and external perturbations, implying

$$\lambda e_p + e_v = 0, \tag{26}$$

and

$$\dot{e}_p = -\lambda e_p, \tag{27}$$

which corresponds to an exponentially stable system. The stability of the SMC can be analysed using the Lyapunov theory for stability, from which the control law is defined. [18] explains that, with the Lyapunov candidate function

$$V = \frac{1}{2} s^T s, \tag{28}$$

for a $\dot{s} = \lambda e_v + \ddot{e}_p$, the Lyapunov derivative must obey

$$\dot{V} \leq -\mu s^T sgn(s) < 0, \tag{29}$$

where $sgn(\ )$ is a function, which returns the sign of its argument, in this case, s. Because $sgn(s)$ is element-wise,

$$s^T sgn(s) = \begin{bmatrix} s_x & s_y & s_z \end{bmatrix} \begin{bmatrix} sgn(s_x) \\ sgn(s_y) \\ sgn(s_z) \end{bmatrix} = |s_x| + |s_y| + |s_z|, \tag{30}$$

and, according to the triangle inequality,

$$||s|| \leq |s_x| + |s_y| + |s_z|. \tag{31}$$

Thus,

$$||s|| \leq s^T sgn(s). \tag{32}$$

Defining $W = \sqrt{2V} = ||s||$, the upper right hand derivative of W, using the chain rule,

$$D^+ W = \frac{1}{||s||} \dot{V} \leq -\mu, \tag{33}$$

and the comparison lemma allow that

$$W(s(t)) \leq W(s(0)) - \mu t, \tag{34}$$

proving that the trajectories reach the surface, $s = 0$, in finite time

$$t \leq \frac{W(s(0))}{\mu}. \tag{35}$$

Finally, (29) proves the trajectories stay in the surface.

The popularity of SMC is due to the fact that, in the sliding surface, the system is invariant to model uncertainties and external disturbances, because, as it is explained, in theory, after reaching the sliding surface the system is confined to it. However, in reality, the system sliding along the surface yields high frequency oscillations [19], corresponding to the chattering effect, which may be reduced by using an approximation of the sign function [20], the boundary layer method [21], the second order SMC [22], or terminal SMC [23].

IV. Design and Implementation of the Trajectory Tracking Controllers for the PX4 Autopilot and Gazebo Simulation

The controllers designed, and implemented in Simulink, for the PX4 Autopilot and Gazebo simulation were the PD, PID, BS, SM, and adaptive BS. The final designs of the PD and BS controllers have an identical control law. In fact, the gains of both controllers may be adjusted so that the response of the system is similar, and the same reasoning can be applied to the Integral BS and PID controllers. Therefore, it is only shown the design of the BS, PID, and SM position controllers.

A. BS Controller

First, the position error,

$$z_1 = p - p_d, \tag{36}$$

and the first Lyapunov candidate function,

$$V_1(z_1) = \frac{1}{2} z_1^T z_1 \succ 0, \tag{37}$$

are defined. The derivative of $V_1(z_1)$,

$$\dot{V}_1(z_1) = z_1^T \dot{z}_1, \tag{38}$$

must be negative definite, for all $z_1 \neq 0$, in order to ensure asymptotic stability in the sense of Lyapunov. The derivative may be rewritten as

$$\dot{V}_1(z_1) = -k_1 z_1^T z_1 + z_1^T (k_1 z_1 + \dot{z}_1), \tag{39}$$

in which k_1 is a positive constant. Then, $\dot{V}_1 \prec 0$ if $k_1 z_1 + \dot{z}_1 = 0$, defining the second error,

$$z_2 = k_1 z_1 + \dot{z}_1, \tag{40}$$

and its derivative

$$\dot{z_2} = k_1 z_2 - k_1^2 z_1 - \frac{T}{m} r_3 + g e_3 - \ddot{p}_d. \tag{41}$$

The second Lyapunov candidate function is set as

$$V_2(z_1, z_2) = k_1^2 V_1(z_1) + \frac{1}{2} z_2^T z_2 \succ 0. \tag{42}$$

The derivative of $V_2(z_1, z_2)$,

$$\dot{V}_2(z_1, z_2) = k_1^2 \dot{V}_1(z_1) + z_2^T \dot{z_2}, \tag{43}$$

must be negative definite. To do so, a virtual control of the desired thrust, T_d, and orientation, r_{3d}, of the quadrotor,

$$u = \frac{T_d}{m} r_{3d}, \tag{44}$$

is created. Considering (39) and (41), (43) may be rewritten as follows,

$$\begin{aligned} \dot{V}_2(z_1, z_2) = -k_1^3 z_1^T z_1 + \\ z_2^T \left(-\frac{T_d}{m} r_{3d} + g e_3 - \ddot{p}_d + k_1 z_2 \right) + \\ z_2^T \left(-\frac{T}{m} r_3 + \frac{T_d}{m} r_{3d} \right). \end{aligned} \tag{45}$$

Then, for $\dot{V}_2$ to be negative definite,

$$-\frac{T}{m}r_3 + \frac{T_d}{m}r_{3d} = 0, \tag{46}$$

and

$$-\frac{T_d}{m}r_{3d} + ge_3 - \ddot{p}_d + k_1 z_2 = -k_2 z_2, \tag{47}$$

with $k_2 > 0$. In order to determine the third error, z_3, it becomes necessary to simplify (46). By projecting the desired virtual control along the direction of the real thrust, as shown in Figure 2, if

$$T = T_d r_3^T r_{3d}, \tag{48}$$

it is concluded, for $T = T_d$ and $r_3 = r_{3d}$,

$$\frac{T_d}{m}r_{3d} = \frac{T}{m}r_3. \tag{49}$$

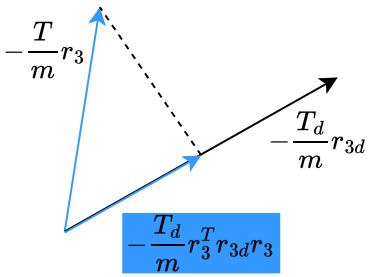


Fig. 2. Projection of the desired virtual control into the the real virtual control.

Then (46) becomes

$$\frac{T_d}{m}S(r_3)^2 z_3 = 0, \tag{50}$$

with $z_3 = r_3 - r_{3d}$ and $-S(r_3)^2$ being the orthogonal projection to r_3. Note that $S(\cdot)$ corresponds to the skew-symmetric matrix defined in (6). From (47), the virtual control can be expressed as

$$u = \frac{T_d}{m}r_{3d} = (k_2 + k_1)z_2 + ge_3 - \ddot{p}_d, \tag{51}$$

resulting in

$$\dot{V}_2(z_1, z_2) = -k_1^3 z_1^T z_1 - k_2 z_2^T z_2 + z_2^T \frac{T_d}{m}S(r_3)^2 z_3, \tag{52}$$

which is negative definite if $z_3 = 0$.

The actuation, T, is derived using (51) and (48), considering

$$\frac{T_d}{m} = \|u\|, \tag{53}$$

in which, $\|.\|$ is the Euclidean Norm. The desired orientation corresponds to

$$r_{3d} = \frac{u}{\|u\|}. \tag{54}$$

The control law for the torque, n, is designed by continuing the backstepping procedure. The third Lyapunov candidate function corresponds to

$$V_3(z_1, z_2, z_3) = V_2(z_1, z_2) + \frac{1}{2}z_3^T z_3 \succ 0, \tag{55}$$

and the function derivative,

$$\dot{V}_3(z_1, z_2, z_3) = \dot{V}_2(z_1, z_2) + z_3^T \dot{z}_3, \tag{56}$$

must be negative definite. Let

$$\dot{V}_3(z_1, z_2, z_3) = -k_1^3 z_1^T z_1 - k_2 z_2^T z_2 + z_3^T \left(\frac{T_d}{m}S(r_3)^2 z_2 + \dot{r}_3 - \dot{r}_{3d}\right), \tag{57}$$

be the expansion of (56). From (5) and (54), the derivatives of the actual and desired orientations, r_3 and r_{3d}, are

$$\dot{r}_3 = -S(r_3)R\omega = -S(r_3)R\Pi_{e_3}\omega, \tag{58}$$

and

$$\dot{r}_{3d} = -\frac{1}{\|u\|}S(r_{3d})^2 \dot{u} = -S(r_{3d})^2 \dot{r}_{3d}, \tag{59}$$

in which $-S(r_{3d})^2$ is the orthogonal projection to r_{3d}. Particularly in (58), it becomes evident that ω_3 will not influence the actuation of the system. The desired angular velocity, ω_d, is included in (57) as $S(r_3)R\Pi_{e_3}\omega_d$, defining the final error, $z_4 = \Pi_{e_3}(\omega - \omega_d)$. Then,

$$\dot{V}_3(z_1, z_2, z_3) = -k_1^3 z_1^T z_1 - k_2 z_2^T z_2 + z_3^T \left(\frac{T_d}{m}S(r_3)^2 z_2 - S(r_3)R\Pi_{e_3}\omega_d + S(r_{3d})^2 \dot{r}_{3d}\right) - z_3^T S(r_3)R z_4. \tag{60}$$

The desired angular velocity is calculated so that

$$z_3^T \left(\frac{T_d}{m}S(r_3)^2 z_2 - S(r_3)R\Pi_{e_3}\omega_d + S(r_{3d})^2 \dot{r}_{3d}\right) \preceq 0. \tag{61}$$

Thus,

$$\Pi_{e_3}\omega_d = \Pi_{e_3} R^T \left(\frac{T_d}{m}S(r_3)z_2 + S(r_{3d})\dot{r}_{3d} - k_3 S(r_3)z_3\right), \tag{62}$$

with $k_3 > 0$, and $\dot{V}_3$ becomes,

$$\dot{V}_3(z_1, z_2, z_3) = -k_1^3 z_1^T z_1 - k_2 z_2^T z_2 - k_3 z_3^T S(r_3)S(r_3)^T z_3 - z_3^T S(r_3)R z_4, \tag{63}$$

being negative definite when $z_4 = 0$. Following this, the fourth and last Lyapunov candidate function is

$$V_4(z_1, z_2, z_3, z_4) = V_3(z_1, z_2, z_3) + \frac{1}{2}z_4^T z_4 \succ 0. \tag{64}$$

The final control law is formulated by guaranteeing that the derivative of V_4,

$$\dot{V}_4(z_1, z_2, z_3, z_4) = \dot{V}_3(z_1, z_2, z_3) + z_4^T \dot{z}_4, \tag{65}$$

is negative semi-definite. This is achieved by expanding the previous equation into

$$\dot{V}_4(z_1, z_2, z_3, z_4) = -k_1^3 z_1^T z_1 - k_2 z_2^T z_2 - k_3 z_3^T S(r_3)S(r_3)^T z_3 + z_4^T (R^T S(r_3)z_3 + \Pi_{e_3}(\dot{\omega} - \dot{\omega}_d)), \tag{66}$$

and choosing $\dot{\omega}$ as

$$\Pi_{e_3}\dot{\omega} = \Pi_{e_3}\left(-R^T S(r_3)z_3 - k_4 z_4 + \dot{\omega}_d\right). \tag{67}$$

Thus, the derivative of V_4 becomes

$$\dot{V}_4(z_1,z_2,z_3,z_4) = -k_1^3 z_1^T z_1 - k_2 z_2^T z_2 - k_3 z_3^T S(r_3) S(r_3)^T z_3 - k_4 z_4^T z_4 \preceq 0, \quad (68)$$

in which $k_4 > 0$. Finally, from (14) and (67), the control law for the torque is

$$n = J\left(-R^T S(r_3) z_3 - k_4 z_4 + \dot{\omega}_d\right) + S(\omega) J \omega. \quad (69)$$

B. PID Controller

Let z_i be the integral of the position error (36), z_1,

$$z_i = \int z_1. \quad (70)$$

In order to prove the stability of the PID controller, it is defined a state vector, x, which contains the errors, z_i, z_1 and z_2 (derivative of z_1),

$$x = \begin{bmatrix} z_i \\ z_1 \\ z_2 \end{bmatrix}, \quad (71)$$

with derivative,

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -k_i & -k_1 & -k_2 \end{bmatrix} x = Ax. \quad (72)$$

The following Lyapunov Candidate function is proposed,

$$V(x) = x^T P x, \quad (73)$$

in which P is a symmetric positive definite matrix. The time derivative of $V(x)$ is

$$\dot{V}(x) = x^T (A^T P + PA) x = -x^T Q x, \quad (74)$$

and, if $Q \succeq 0$ and A is stable, then there exists a solution for P and the system is stable in the sense of Lyapunov.

Then, it is necessary to include the orientation error, z_3. From the previous designs of the controllers, it is known that

$$\dot{z}_2 = -\frac{T_d}{m} r_{3d} + g e_3 - \ddot{p}_d + \frac{T_d}{m} S(r_3)^2 z_3. \quad (75)$$

Following this train of thought (72) becomes

$$\dot{x} = Ax + \begin{bmatrix} 0 \\ 0 \\ \frac{T_d}{m} S(r_3)^2 z_3 \end{bmatrix} = Ax + B(T_d, r_3, z_3), \quad (76)$$

changing the time derivative of the Lyapunov function to

$$\dot{V}(x) = -x^T Q x + 2 x^T P B(T_d, r_3, z_3). \quad (77)$$

The third and last Lyapunov Candidate function is

$$V(x) = x^T P x + \frac{1}{2} z_3^T z_3, \quad (78)$$

which has a derivative of,

$$\dot{V}(x) = -x^T Q x + 2 x^T P B(T_d, r_3, z_3) + z_3^T \dot{z}_3. \quad (79)$$

The orientation error has a derivative of

$$\dot{z}_3 = -S(r_3) R \Pi_{e_3} \omega + S(r_{3d})^2 \dot{r}_{3d}. \quad (80)$$

In order for (79) positive semi definite, the angular velocity, ω is defined as

$$\Pi_{e_3} \omega = \Pi_{e_3} R^T \left(S(r_{3d}) \dot{r}_{3d} - k_3 S(r_3) z_3\right). \quad (81)$$

C. SM Controller

First, it is necessary to define a sliding surface, $s = 0$. Much like in the previous section, let z_1 be the position error, and z_2 and $\dot{z}_2$ be its first and second derivatives. The sliding surface and its derivative are defined as

$$s(z_1, z_2) = \lambda z_1 + z_2 = 0, \quad (82)$$

$$\dot{s}(z_1, z_2) = \lambda z_2 + \dot{z}_2 = 0, \quad (83)$$

in which λ is a positive constant. Considering the sliding surface, let the Lyapunov candidate function be

$$V(s(z_1, z_2)) = \frac{1}{2} s^T(z_1, z_2) s(z_1, z_2) \succ 0. \quad (84)$$

Its time derivative is

$$\dot{V}(s(z_1, z_2)) = s^T(z_1, z_2) \dot{s}(z_1, z_2), \quad (85)$$

can, using (83), be rewritten as

$$\dot{V}(s(z_1, z_2)) = s^T(z_1, z_2) \left(\lambda z_2 - \frac{T}{m} r_3 + g e_3 - \ddot{p}_d\right) \quad (86)$$

by expanding $\dot{s}$ and, by including the virtual control (44), it becomes

$$\dot{V}(s(z_1, z_2)) = s^T(z_1, z_2) \left(\lambda z_2 - \frac{T_d}{m} r_{3d} + g e_3 - \ddot{p}_d + \frac{T_d}{m} S(r_3)^2 z_3\right), \quad (87)$$

in which z_3 is the orientation error. From (87), it is concluded that

$$\dot{s}(z_1, z_2) = \lambda z_2 - \frac{T_d}{m} r_{3d} + g e_3 - \ddot{p}_d + \frac{T_d}{m} S(r_3)^2 z_3. \quad (88)$$

In order to reduce the chattering effect, the element-wise hyperbolic tangent function [24], [25] is applied instead of the sgn function,

$$\dot{s}_d = -\mu \tanh\left(s(z_1, z_2)\right). \quad (89)$$

Following (89), the virtual control law corresponds to

$$\frac{T_d}{m} r_{3d} = \lambda z_2 + g e_3 - \ddot{p}_d + \mu \tanh\left(s(z_1, z_2)\right), \quad (90)$$

and the real thrust is calculated using (48).

Thus, (87) becomes,

$$\dot{V}(s(z_1, z_2)) \leq s^T(z_1, z_2) \left(-\mu \tanh\left(s(z_1, z_2)\right) + \frac{T_d}{m} S(r_3)^2 z_3\right). \quad (91)$$

The hyperbolic tangent function is in the odd quadrants, which means $-s^T tanh(s)$ is always negative. Then, $\dot{V}(s(z_1, z_2))$ is negative definite, when $z_3 = 0$.

The desired angular velocity and the torque actuation are obtained through the backstepping procedure, which is previously explained.

Controller	k_1	k_2	k_i	k_3
BS	1	6	-	7
PD	5	10	-	7
PID	5	10	5	7

TABLE I
GAINS OF THE CONTROLLERS.

Trajectory	Controller	MAE [m]		
		z_{1x}	z_{1y}	z_{1z}
Hover	PD	0.01	0.01	0.2
	PID	0.007	0.01	0.02
	BS	0.02	0.03	0.3
	SM	0.01	0.02	0.1
Circular	PD	0.03	0.01	0.2
	PID	0.04	0.01	0.02
	BS	0.07	0.02	0.3
	SM	0.04	0.02	0.08

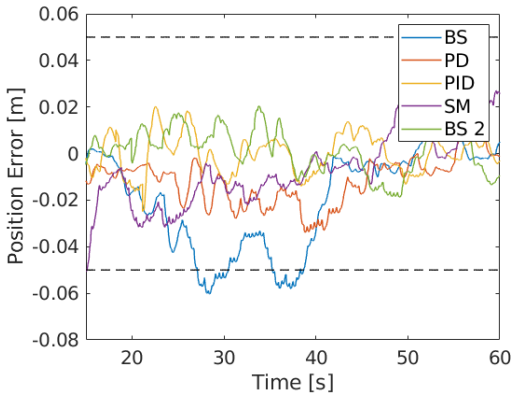
TABLE II
MAE OF THE x, y AND z POSITION ERRORS, FOR BOTH CIRCULAR TRAJECTORIES.

V. SIMULATION RESULTS

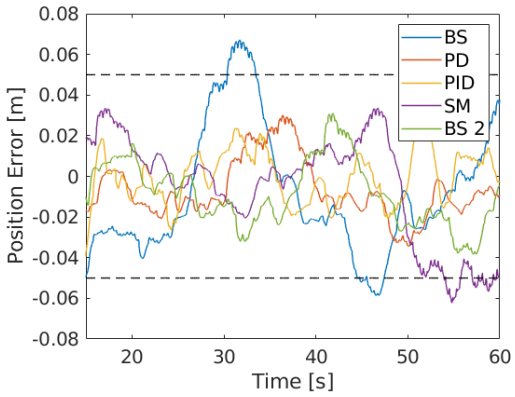
After implementing the controllers in Simulink, the respective gains were tuned so that the vehicle could follow velocity and position references. The gains were selected based on the time it took for the quadrotor to achieve a margin of error of 15%, defining the achieved time, AT. Table I displays the gains chosen for the BS, PD, and PID controllers. For the SM controller, it was defined $\lambda = 5$, and $\mu = 5$. The evaluation of the controllers was based on position tracking and the time it took for the error response to settle within a 5% margin, referred to as settling time, ST. Four different trajectories were tested: hover, ascending, circular, and circular while ascending. In this paper, only the results of the hover and circular experiments are shown.

Figures 3 and 4 illustrate the position error evolution throughout time, for the hover and circular trajectory, respectively.

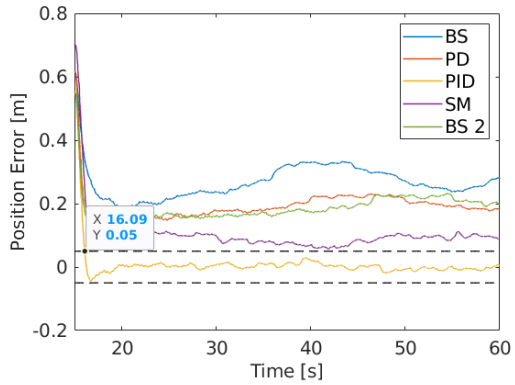
The sum of the ST of each coordinate of the PID controller consists of $1\,\mathrm{s}$ and $8\,\mathrm{s}$ for the hover and circular experiments, respectively. The ST of the PD, SM, and BS controllers is not presented because, for these cases, the altitude error never reaches or stays within the 5% margin. In addition to analysing the ST of the different controllers, the metrics IAE and MAE were used to compare the position errors. A reduced IAE and MAE means that the controller has a good performance, tracking the desired trajectory over time. Among all the simulated controllers, the BS solution exhibits the highest IAE and MAE for the hover and circular experiments. Therefore, the BS solution demonstrates the poorest performance when compared to the other controllers. In terms of the altitude error, the PID controller exhibits the lowest IAE and MAE, which is reasonable because of the integral action. The PID MAE decreased 93%, and 90% compared to the BS, and PD, in all trajectories. The PID altitude IAE surpasses alternative control strategies by an order of magnitude. The SM controller possesses the lowest altitude IAE among the non-integral controllers. The x and y MAE and IAE for both trajectories is similar among the studied controllers, having the same order of magnitude.



(a) x position.



(b) y position.

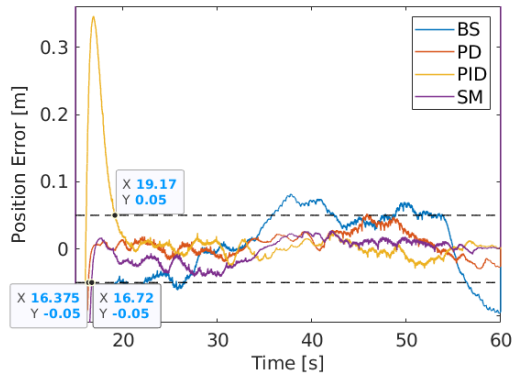


(c) z position.

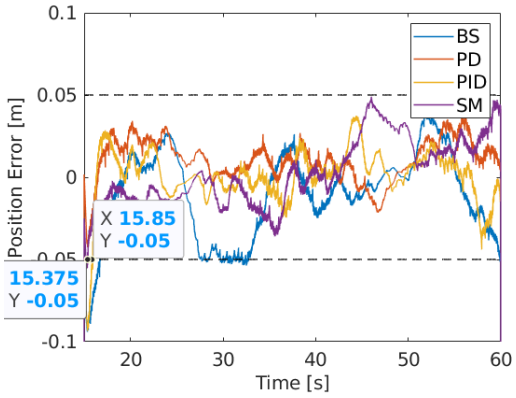
Fig. 3. Hover Trajectory: Position error, z_1. BS 2 corresponds to the Backstepping controller which is tuned according to the PD gains. BS is the original Backstepping controller.

Trajectory	Controller	IAE [m s]		
		z_{1x}	z_{1y}	z_{1z}
Hover	PD	0.58	0.63	8.68
	PID	0.31	0.48	0.71
	BS	0.98	1.22	12.04
	SM	0.64	0.91	4.39
Circular	PD	1.39	0.63	8.26
	PID	1.67	0.50	0.65
	BS	3.14	0.97	12.94
	SM	1.62	0.75	3.67

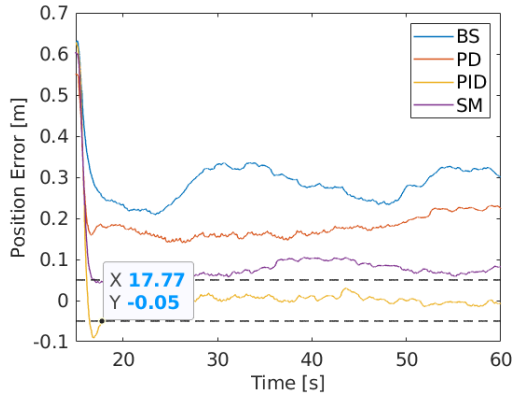
TABLE III
IAE OF THE x, y AND z POSITION ERRORS, FOR BOTH CIRCULAR TRAJECTORIES.



(a) x position.

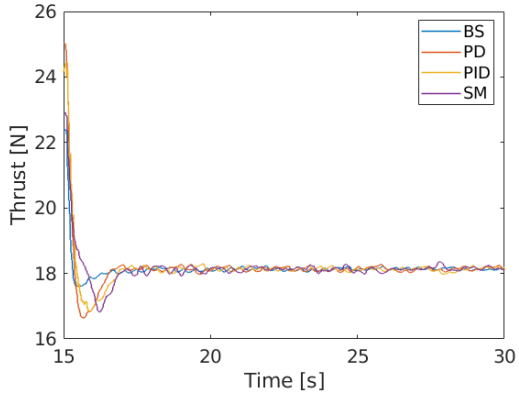


(b) y position.

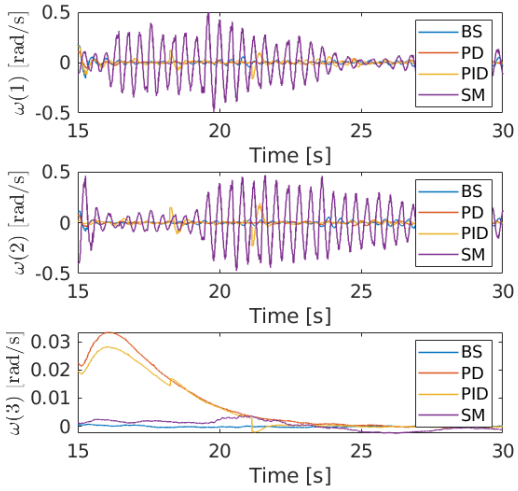


(c) z position.

Fig. 4. Circular Trajectory: Position Error, z_1.



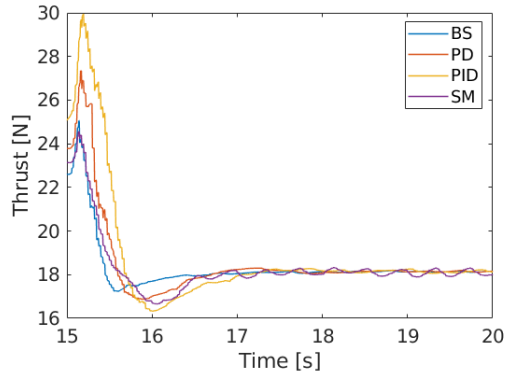
(a) Thrust.



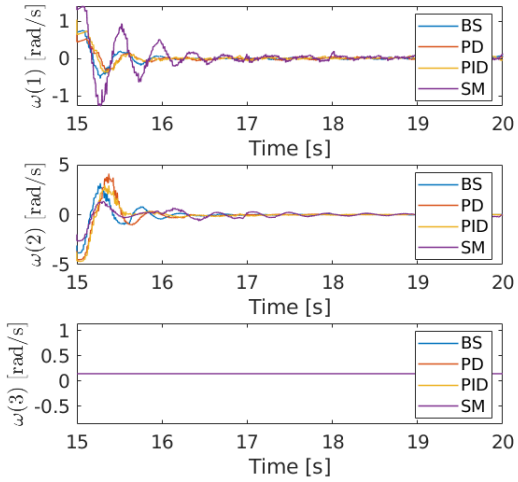
(b) Angular Rates Reference.

Fig. 5. Hover Trajectory, Actuation.

Additionally, Figures 5 and 6 show the actuation, thrust and angular velocity reference. The thrust value for both trajectories is identical, stabilising at, approximately, $18\,\mathrm{N}$. The angular rates of the SM controller present high frequency oscillations, in both trajectories, which may be a result of the chattering effect. Finally, Figure 7 illustrates the position error of the quadrotor when the vehicle is commanded to follow a certain trajectory in a windy environment. In these experiments, both the PID and SM controllers showed the best results, with the altitude and x position errors close to zero. However, only the PID controller was able to keep the x position error within the 5% error margin. Nonetheless, it was possible to prove the robustness of the SM controller when there existed environment disturbances in comparison with the non-integral controllers. Tables IV and V display the values of the position MAE and IAE of the different controllers. The findings make it possible to conclude that the MAE of the x position increased, in comparison to the values in Tables II and III, and the PID continues to present the lowest values of all the approaches. On the other hand, the SM also has a reduced MAE, considering only the non-integral controllers. Both PID and SM have an altitude MAE in the order of $10^{-2}\mathrm{m}$, while the PD and BS MAE falls within the range of $10^{-1}\mathrm{m}$. This shows the good performance of the PID and SM controllers in rejecting the wind influence. As expected the IAE of the x coordinate increased for all the presented situations. Once again, the PID controller demonstrates a good performance, in terms of rejecting the wind perturbation, because the value of the respective IAE is the lowest of all the controllers. The SM controller also presents a good performance, having the lowest IAE value compared to the PD and BS controllers.



(a) Thrust.



(b) Angular Rates Reference.

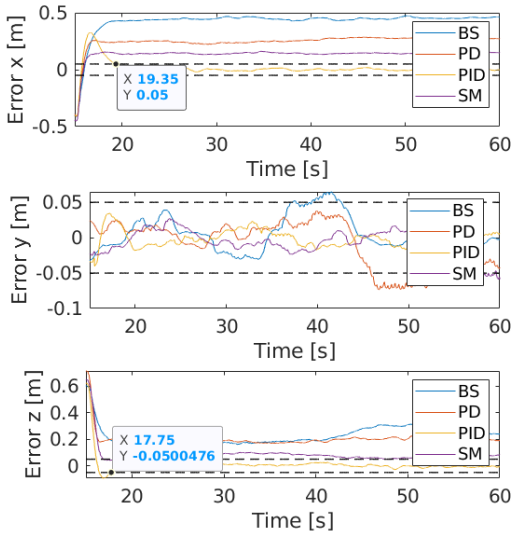
Fig. 6. Circular Trajectory: Actuation.

Trajectory	Controller	MAE [m]		
		z_{1x}	z_{1y}	z_{1z}
Hover	PD	0.26	0.03	0.21
	PID	0.03	0.01	0.02
	BS	0.43	0.02	0.22
	SM	0.15	0.01	0.08
Circular	PD	0.29	0.02	0.24
	PID	0.06	0.01	0.03
	BS	0.45	0.04	0.23
	SM	0.18	0.01	0.07

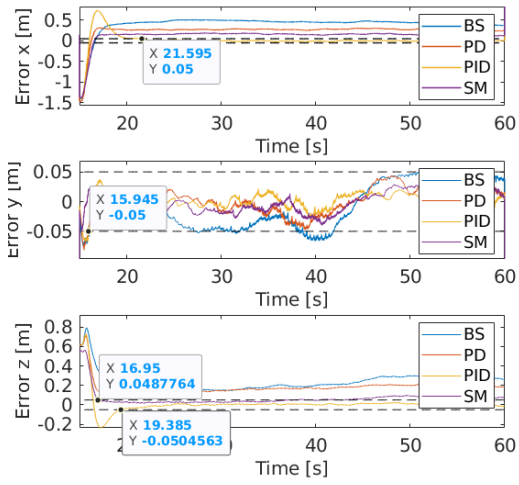
TABLE IV
WIND SIMULATION: MAE OF THE x, y AND z POSITION ERRORS, FOR THE HOVER AND CIRCULAR TRAJECTORIES.

Trajectory	Controller	IAE [m s]		
		z_{1x}	z_{1y}	z_{1z}
Hover	PD	11.67	1.31	9.27
	PID	1.16	0.45	0.96
	BS	19.55	1.01	10.07
	SM	6.61	0.67	3.62
Circular	PD	12.89	0.85	8.16
	PID	2.88	0.45	1.55
	BS	20.27	1.65	10.30
	SM	7.91	0.67	2.97

TABLE V
WIND SIMULATION: IAE OF THE x, y AND z POSITION ERRORS, FOR THE HOVER AND CIRCULAR TRAJECTORIES.



(a) Hover.



(b) Circular.

Fig. 7. Wind Simulation: Position Error.

VI. ADAPTIVE CONTROL DESIGN AND IMPLEMENTATION

The adaptive controller was elaborated assuming the unknown variable was the inverse of the mass of the quadrotor, $\alpha = \frac{1}{m}$, which is a scalar, and considering only the outer loop of the control system of the quadrotor. The adaptive controller was simulated in the Simulink which includes the drone interface, and the actuation was the thrust and the Euler angles. To simulate the variation of the mass of the quadrotor it was made a simple switch that at $55\,\mathrm{s}$ would change the initial condition of the integrator which received the control law. For this experiment the quadrotor was set to hover at $2\,\mathrm{m}$. The results of the simulation are presented in Figure 8. The moment of variation of the mass is noticeable by observing the evolution of the altitude error. After the initial condition is changed, the error increases, but stabilises at zero $13\,\mathrm{s}$ later.

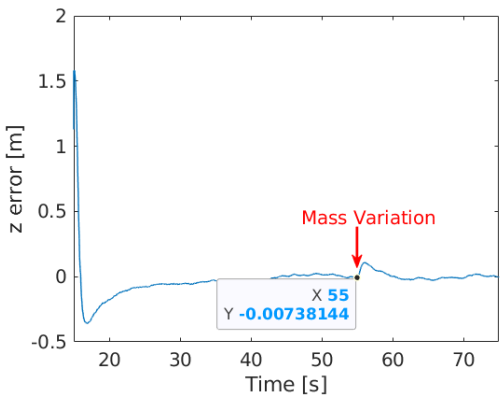


Fig. 8. Adaptive BS controller: Altitude Error.

VII. Conclusion

This thesis proposed the design, implementation, and evaluation of trajectory tracking controllers for the quadrotor UAV. After studying a number of approaches typically used for this control scheme, PD, PID, BS, SM and adaptive backstepping controllers were designed and tested using MATLAB Simulink, the PX4 Autopilot, and the Gazebo environment simulator. For the evaluation of the controllers, this paper compared the values of the ST, IAE, and MAE. The controllers were evaluated by following four different trajectories: hover, ascending, circular, and circular while ascending. However, this work only shows the results for two trajectories, hover and circular. As expected, the PID controller demonstrated the best position tracking performance, as the altitude error remained within the 5% margin in every situation, while the other controllers, lacking integral action, never settled within the predefined margin. Additionally, the ST, IAE, and MAE of the PID controller were the lowest of all the controllers. The controllers were also tested in an environment influenced by wind forces. In these experiments, both the PID and SM controllers showed better results, with altitude and x position errors close to zero, and the lowest MAE and IAE. Of all the non-integral controllers, the SM proved to have the best tracking performance, and possibly could even surpass the PID results if there were added an integral term, by tuning the gains more precisely, and if the chattering phenomenon could be more reduced, as the angular velocity showed signs of chattering. Finally, an adaptive controller was implemented, which proved to maintain the desired height even when there was a sudden variation in the mass estimate.

References

[1] B. J. Emran and H. Najjaran, "A review of quadrotor: An underactuated mechanical system," *Annual Reviews in Control*, vol. 46, pp. 165–180, 2018.

[2] H. Mo and G. Farid, "Nonlinear and adaptive intelligent control techniques for quadrotor uav–a survey," *Asian Journal of Control*, vol. 21, no. 2, pp. 989–1008, 2019.

[3] Y.-R. Tang and Y. Li, "Dynamic modeling for high-performance controller design of a uav quadrotor," in *2015 IEEE international conference on information and automation*. IEEE, 2015, pp. 3112–3117.

[4] Y. Li and C. Liu, "Applications of multirotor drone technologies in construction management," *International Journal of Construction Management*, vol. 19, no. 5, pp. 401–412, 2019.

[5] 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.

[6] N. S. Özbek, M. Önkol, and M. Ö. Efe, "Feedback control strategies for quadrotor-type aerial robots: a survey," *Transactions of the Institute of Measurement and Control*, vol. 38, no. 5, pp. 529–554, 2016.

[7] 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)*, vol. 3. IEEE, 2004, pp. 2451–2456.

[8] J. Li and Y. Li, "Dynamic analysis and pid control for a quadrotor," in *2011 IEEE International Conference on Mechatronics and Automation*. IEEE, 2011, pp. 573–578.

[9] A. L. Salih, M. Moghavvemi, H. A. Mohamed, and K. S. Gaeid, "Modelling and pid controller design for a quadrotor unmanned air vehicle," in *2010 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, vol. 1. IEEE, 2010, pp. 1–5.

[10] H. K. Khalil, *Nonlinear systems; 3rd ed.* Upper Saddle River, NJ: Prentice-Hall, 2002, the book can be consulted by contacting: PH-AID: Wallet, Lionel. [Online]. Available: https://cds.cern.ch/record/1173048

[11] T. Madani and A. Benallegue, "Backstepping control for a quadrotor helicopter," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 3255–3260.

[12] M. A. Mohd Basri, K. A. Danapalasingam, and A. R. Husain, "Design and optimization of backstepping controller for an underactuated autonomous quadrotor unmanned aerial vehicle," *Transactions of FAMENA*, vol. 38, no. 3, pp. 27–44, 2014.

[13] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, pp. 153–158.

[14] A. A. Mian and W. Daobo, "Modeling and backstepping-based nonlinear control strategy for a 6 dof quadrotor helicopter," *Chinese Journal of Aeronautics*, vol. 21, no. 3, pp. 261–268, 2008.

[15] M. Huang, B. Xian, C. Diao, K. Yang, and Y. Feng, "Adaptive tracking control of underactuated quadrotor unmanned aerial vehicles via backstepping," in *Proceedings of the 2010 American Control Conference*. IEEE, 2010, pp. 2076–2081.

[16] D. Cabecinhas, R. Cunha, and C. Silvestre, "A globally stabilizing path following controller for rotorcraft with wind disturbance rejection," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, pp. 708–714, 2015.

[17] Z. Fang and W. Gao, "Adaptive integral backstepping control of a micro-quadrotor," in *2011 2nd International conference on intelligent control and information processing*, vol. 2. IEEE, 2011, pp. 910–915.

[18] K. K. Hassan *et al.*, "Nonlinear systems," *Departement of Electrical and computer Engineering, Michigan State University*, 2002.

[19] J. Samantaray and S. Chakrabarty, "Discrete time sliding mode control," in *Control Theory in Engineering*, C. Volosencu, A. Saghafinia, X. Du, and S. Chakrabarty, Eds. Rijeka: IntechOpen, 2020, ch. 5. [Online]. Available: https://doi.org/10.5772/intechopen.91245

[20] R. Xu and U. Ozguner, "Sliding mode control of a quadrotor helicopter," in *Proceedings of the 45th IEEE Conference on Decision and Control*. IEEE, 2006, pp. 4957–4962.

[21] K. Runcharoon and V. Srichatrapimuk, "Sliding mode control of quadrotor," in *2013 The International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE)*. IEEE, 2013, pp. 552–557.

[22] M. Bouchoucha, S. Seghour, and M. Tadjine, "Classical and second order sliding mode control solution to an attitude stabilization of a four rotors helicopter: From theory to experiment," in *2011 IEEE International Conference on Mechatronics*. IEEE, 2011, pp. 162–169.

[23] M. Labbadi and M. Cherkaoui, "Robust integral terminal sliding mode control for quadrotor uav with external disturbances," *International Journal of Aerospace Engineering*, vol. 2019, 2019.

[24] A. Noordin, M. A. Mohd Basri, Z. Mohamed, and I. Mat Lazim, "Position and attitude control of quadrotor mav using sliding mode control with tanh function," in *Enabling Industry 4.0 through Advances in Mechatronics: Selected Articles from iM3F 2021, Malaysia*. Springer, 2022, pp. 193–204.

[25] M. Labbadi, M. Cherkaoui, Y. E. houm, and M. Guisser, "Modeling and robust integral sliding mode control for a quadrotor unmanned aerial vehicle," in *2018 6th International Renewable and Sustainable Energy Conference (IRSEC)*, 2018, pp. 1–6.