POLITECNICO DI TORINO
Repository ISTITUZIONALE

A Comparative Study for Control of Quadrotor UAVs

Original
A Comparative Study for Control of Quadrotor UAVs / Rinaldi, Marco; Primatesta, Stefano; Guglieri, Giorgio. - In: APPLIED SCIENCES. - ISSN 2076-3417. - 13:6(2023), p. 3464. [10.3390/app13063464]

Availability:
This version is available at: 11583/2976679 since: 2023-03-09T11:20:42Z

Publisher:
MDPI

Published
DOI:10.3390/app13063464

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

12 October 2024

applied sciences

MDPI

Article

A Comparative Study for Control of Quadrotor UAVs

Marco Rinaldi, Stefano Primatesta and Giorgio Guglieri *

Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy
* Correspondence: giorgio.guglieri@polito.it

Abstract: Modeling and controlling highly nonlinear, multivariable, unstable, coupled and underactuated systems are challenging problems to which a unique solution does not exist. Modeling and control of Unmanned Aerial Vehicles (UAVs) with four rotors fall into that category of problems. In this paper, a nonlinear quadrotor UAV dynamical model is developed with the Newton–Euler method, and a control architecture is proposed for 3D trajectory tracking. The controller design is decoupled into two parts: an inner loop for attitude stabilization and an outer loop for trajectory tracking. A few attitude stabilization methods are discussed, implemented and compared, considering the following control approaches: Proportional–Integral–Derivative (PID), Linear–Quadratic Regulator (LQR), Model Predictive Control (MPC), Feedback Linearization (FL) and Sliding Mode Control (SMC). This paper is intended to serve as a guideline work for selecting quadcopters' control strategies, both in terms of quantitative and qualitative considerations. PID and LQR controllers are designed, exploiting the model linearized about the hovering condition, while MPC, FL and SMC directly exploit the nonlinear model, with minor simplifications. The fast dynamics ensured by the SMC-based controller together with its robustness and the limited estimated command effort of the controller make it the most promising controller for quadrotor attitude stabilization. The outer loop consists of three independent PID controllers: one for altitude control and the other two, together with a dynamics' inversion, are entitled to the computation of the reference attitude for the inner loop. The capability of the controlled closed-loop system of executing complex trajectories is demonstrated by means of simulations in MATLAB/Simulink®.

Keywords: unmanned aerial vehicles; feedback linearization; sliding mode control; MPC; PID; LQR; UAV modeling; UAV control; attitude stabilization; trajectory tracking; performance evaluation

Citation: Rinaldi, M.; Primatesta, S.; Guglieri, G. A Comparative Study for Control of Quadrotor UAVs. *Appl. Sci.* **2023**, *13*, 3464. https://doi.org/10.3390/app13063464

Academic Editor: Jérôme Morio

Received: 15 February 2023
Revised: 3 March 2023
Accepted: 7 March 2023
Published: 8 March 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The growth of the number of applications of Unmanned Aerial Vehicles (UAVs) in many different fields, from civilian to military applications, has experienced a big rise over the last decade. As highlighted in specific literature reviews [1,2], the civilian applications of UAVs include activities such as parcel delivery, transportation of medical samples, mapping, surveillance, precision agriculture and greenhouses [3], population health monitoring, disaster recovery, personal use and photography, to name a few. The advantages due to the adoption of UAVs in each application domain have been widely reviewed. Just to give an example, considering the logistic domain discussed in [4], UAVs can contribute to optimizing the logistic process, reducing resources' usage, traffic and CO_2 emissions, inducing the diversification of transportation options as well as giving rise to new business models.

Therefore, UAVs have gained significant interest from the engineering research world over the last two decades. In particular, quadrotors are the most researched and used type of UAVs because of their mechanical structure, which is relatively simple to model, their enhanced closed-loop stability with respect to other configurations and their high maneuverability in both indoor and outdoor spaces, as described in [5]. However,

quadcopters are mechatronic systems with nonlinear, multivariable, unstable, coupled and underactuated characteristics, which make the control not trivial.

Literature reviews such as [6–9] provide a comprehensive analysis of the main state-of-the-art control strategies for quadrotor UAVs. The well-known Proportional–Integral–Derivative (PID) control can effectively handle both linear and nonlinear systems. This is because it is model-independent and has a simple-to-understand tuning principle. Additionally, PID control can achieve zero steady-state error. On the other hand, PID control cannot handle constraints, is poorly robust with respect to disturbances and uncertainties and may require extensive tuning. The research in [10] implements a quadrotor attitude controller by means of a PID controller optimized with a genetic algorithm.

The Linear–Quadratic Regulator (LQR) control approach is equivalent to solving an optimization problem, with the cost function weighted of the squared norms of tracking error and control input signals, under the constraint of satisfying the system's dynamics. It is designed with a linear approximation of the system and may not guarantee a zero steady-state error. LQR control can find an optimal feedback gain matrix considering both the tracking performances and the command input effort of the controller.

The Feedback Linearization (FL) control approach consists of a systematic procedure to linearize the system with a model inversion and control it with a new control input term. The system is transformed into a linear system without any linearization assumption and can be controlled according to the linear control theory. FL performs well when the nonlinear plant is similar to the one used for control design, so it is not very robust to parametric uncertainties and cannot handle constraints. A feedback linearization attitude controller with proportional action is proposed in [11].

The Sliding Mode Control (SMC) approach aims at bringing the system state to the "sliding surface", which is a subset of the state space that corresponds to a desired dynamical behavior. The control law is the sum of a continuous function of the system's state and the reference, which guarantees that the "sliding surface" is invariant (once the state is on the sliding surface, it remains on it), and a discontinuous term for convergence to the "sliding surface". SMC shows, in general, good robustness to model uncertainties and disturbances, but a smooth approximation of the control law is necessary to avoid chattering, which may lead to instability.

The Model Predictive Control (MPC) approach solves an optimization problem at each defined time step to compute the optimal control input, satisfying the system's dynamics, input and state constraints. A prediction model is exploited to predict the future states of the system. MPC is quite robust to disturbances but may be slow in tracking. The MPC-based position trajectory controller designed in [12], despite exploiting a linear model for prediction, shows the ability to ensure tracking of complex trajectories.

Furthermore, other control laws can be effectively applied to quadrotors, such as H-infinity, Backstepping, Adaptive, Fuzzy and Neural Network control. A Backstepping controller for position control of a quadrotor helicopter is designed and validated in [13], with stability being proven by the Lyapunov method. Additionally, a Fuzzy Logic controller for attitude control has been proposed in [14]. An L_1 adaptive controller is designed and validated in [15] for the attitude controller of a mini-UAV. Adaptive control does not need a parameter tuning procedure and deals with parametric uncertainties and unmodelled dynamics. The Sliding Mode-based adaptive controller with Proportional–Integral–Derivative gains proposed in [16] controls a micro-UAV, the Parrot Mambo Minidrone, and is demonstrated on a model-based platform. In [17], an adaptive controller is proven to be able to deal with parametric uncertainties and external disturbances and performs well with unknown time delays inserted in the closed-loop dynamics.

In this paper, a comparative study of 3D trajectory control of a quadrotor system is proposed. The control system is decoupled into an outer loop for trajectory control and an inner loop for attitude control; input saturation is also taken into account. Despite

several reviews of control algorithms for quadrotors being proposed in the literature, such as [6–9] to state a few, there is, to the best extent of our knowledge, poor availability of comparative studies where several approaches are implemented and discussed. The comparative study proposed in [18] implements PID, LQR, Fuzzy and Adaptive controllers for position trajectory tracking, while PID, LQR and MPC-based attitude stabilizers are implemented and discussed in [19]. In order to enlarge the available state of the art of quantitative comparative performance evaluation for quadrotors' control, we propose a broad comparative implementation of different control approaches (PID, LQR, MPC, FL and SMC) to the quadrotor attitude stabilization problem. Linearized or simplified attitude dynamical systems are exploited in the design phase, yet all the controllers are tested with the proposed full nonlinear model. According to the fast dynamic response, the robustness to the discrepancy between the control model and the full nonlinear model and an approximative L_2 norm-based trade-off evaluation of both the tracking performance and the command effort of the controller, the most suitable method is selected for the inner loop control part of the complete proposed control architecture. The outer loop is designed to track velocity references by exploiting PID control and a dynamics inversion from a simplified translational dynamical model.

This paper is organized as follows. The system model derived with the Newton–Euler formalism is reported in Section 2, as well as the proposed control system architecture, the proposed control laws and the simulation parameters setting. Section 3 presents the simulation results in which a micro-UAV is demonstrated to be able to execute complex trajectories with the complete control system. Conclusions and future research directions are drawn in Section 4.

2. Materials and Methods

2.1. System Model

Quadrotor UAVs are mechatronic systems with two rigid arms arranged in a cross configuration and four rotors. Each pair of rotors is located at the arm's extremity and rotates in opposite directions. The speed of the rotors is controlled independently. A picture of a commercial quadrotor platform is shown in Figure 1. The movement of flying quadrotors derives from the total lift force produced by all rotors and from their capability of inclination, which is due to the resultant forces acting on the arms and to the resultant counter-torque caused by the pairs of rotors. The representation shown in Figure 2 is derived from the configuration of Figure 1, considered as a reference for the formulation of the quadrotor dynamical model.

Figure 1. Picture of a commercial quadcopter platform [20].

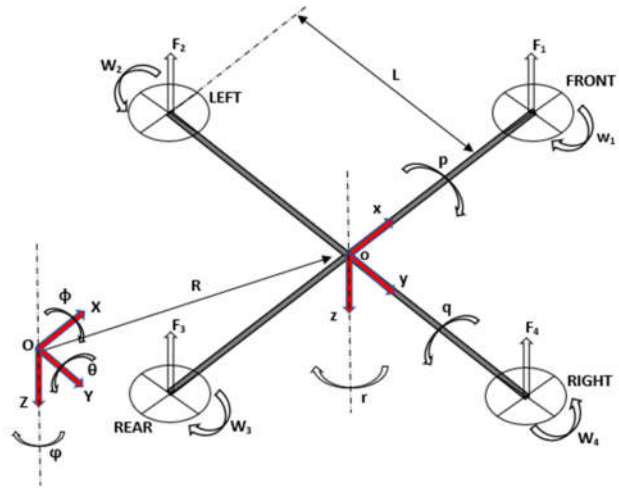

Figure 2. Schematic representation of the quadrotor configuration of reference.

Two reference frames are defined in order to characterize the position and the orientation of the UAV: a fixed inertial reference frame $OXYZ$ and a body reference frame oxyz attached to the quadrotor. The relation between the two is expressed by the homogeneous transformation matrix $\mathbf{T_L}$ derived with the Euler ZYX convention, also known as the RPY convention. The RPY convention involves three rotations. Firstly, a rotation of ψ (Euler yaw angle) about Z produces the frame OX_1Y_1Z. Secondly, a second rotation of ϑ (Euler pitch angle) about Y_1 results in the frame $OX_2Y_1Z_1$. Lastly, a third rotation of ϕ (Euler roll angle) about X_2 results in the frame $OX_2Y_2Z_2$ (coincident with oxyz). These rotations are performed according to a post multiplication rule,

$$\mathbf{T_L} = \mathbf{T_\psi T_\theta T_\phi} = \begin{pmatrix} c_\psi & -s_\psi & 0 \\ s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} c_\theta & 0 & s_\theta \\ 0 & 1 & 0 \\ -s_\theta & 0 & c_\theta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi & c_\phi \end{pmatrix} = \begin{pmatrix} c_\psi c_\theta & c_\psi s_\theta s_\phi - s_\psi c_\phi & s_\psi s_\phi + c_\psi s_\theta c_\phi \\ s_\psi c_\theta & c_\psi c_\phi + s_\theta s_\psi s_\phi & s_\theta s_\psi c_\phi - s_\phi c_\psi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{pmatrix} \tag{1}$$

where c and s denote cosine and sine functions, respectively. $\mathbf{T_L}$ transforms linear movement-related vectors from oxyz to $OXYZ$. Consequently, $\mathbf{T_R}$ transforms rotational movement-related vectors from $OXYZ$ to oxyz,

$$\mathbf{\Delta} = \begin{pmatrix} p \\ q \\ r \end{pmatrix} = \mathbf{T_\phi^T} \begin{pmatrix} \phi` \\ 0 \\ 0 \end{pmatrix} + \mathbf{T_\phi^T T_\theta^T} \begin{pmatrix} 0 \\ \theta` \\ 0 \end{pmatrix} + \mathbf{T_\phi^T T_\theta^T T_\psi^T} \begin{pmatrix} 0 \\ 0 \\ \psi` \end{pmatrix} = \begin{pmatrix} 1 & 0 & -s_\theta \\ 0 & c_\phi & c_\theta s_\phi \\ 0 & -s_\phi & c_\theta c_\phi \end{pmatrix} \begin{pmatrix} \phi` \\ \theta` \\ \psi` \end{pmatrix} = \mathbf{T_R} \begin{pmatrix} \phi` \\ \theta` \\ \psi` \end{pmatrix} \tag{2}$$

where p, q and r denote the rotational velocities in the body frame, while $\phi`$, $\vartheta`$ and $\psi`$ denote the Euler rates. The Newton–Euler formalism is adopted to present the quadrotor dynamical model; for further details about modeling refer to [21–24].

In the adopted model, the following assumptions are made:

- The structure is perfectly symmetrical.
- The structure is perfectly rigid.
- The quadrotor's center of mass coincides with the origin of the body fixed frame.
- The rotors' motors dynamics are neglected.
- Complex phenomena of difficult modeling are neglected, such as the ground effect, blade flapping and any other minor aerodynamical or inertial contributions not explicitly taken into account.

It follows the translational dynamical model in the inertial fixed reference frame,

$$m\mathbf{R}`` = m\mathbf{g} - K_d^L \mathbf{R}` - (\mathbf{F_1} + \mathbf{F_2} + \mathbf{F_3} + \mathbf{F_4})\mathbf{T_L} \tag{3}$$

$$\mathbf{F_i} = \begin{pmatrix} 0 \\ 0 \\ bw_i^2 \end{pmatrix}, \text{with } i = 1,2,3,4 \tag{4}$$

where $\mathbf{R}$ denotes the position vector in $OXYZ$, $\mathbf{F_i}$ is the thrust force produced by rotor i, w_i is the angular speed of rotor i, b is the thrust coefficient, m is the quadcopter's mass, $\mathbf{g}$ is the gravity acceleration and K_d^L denotes the aerodynamic thrust drag coefficient. It follows the rotational dynamical model in the body reference frame,

$$I\mathbf{\Delta}` = -(\mathbf{\Delta} \text{ x } I\mathbf{\Delta}) - \mathbf{M_{gy}} - K_d^R\mathbf{\Delta} + \mathbf{M} \tag{5}$$

$$\mathbf{M_{gy}} = I_R(w_1 + w_3 - w_2 - w_4)\,\mathbf{\Delta} \times \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{6}$$

$$\mathbf{M} = \begin{pmatrix} L(F_2 - F_4) \\ L(F_1 - F_3) \\ d(w_1^2 - w_2^2 + w_3^2 - w_4^2) \end{pmatrix} \tag{7}$$

$$I = \begin{pmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{pmatrix} \tag{8}$$

with the quadrotor's arm length being L, the diagonal inertia matrix I, the inertia of the rotors I_R, the drag coefficient d, the aerodynamic moment drag coefficient K_d^R, the rotors' induced moment vector $\mathbf{M}$ and the gyroscopic moment vector $\mathbf{M_{gy}}$; the latter is due to the fact that the axes of rotation of the propellers are attached to the quadrotor and rotate with it. The complete nonlinear dynamical model of the quadrotor is derived by evaluating Equations (3) and (5) and considering the transformations in Equations (1) and (2):

$$X`` = -\frac{1}{m}K_d^L X` - \frac{1}{m}\left(s_\psi s_\phi + c_\psi s_\theta c_\phi\right)U_1 \tag{9}$$

$$Y`` = -\frac{1}{m}K_d^L Y` - \frac{1}{m}\left(s_\theta s_\psi c_\phi - s_\phi c_\psi\right)U_1 \tag{10}$$

$$Z`` = g - \frac{1}{m}K_d^L Z` - \frac{1}{m}c_\theta c_\phi U_1 \tag{11}$$

$$p` = \frac{I_Y - I_z}{I_x}qr - \frac{I_R}{I_x}q(w_1 + w_3 - w_2 - w_4) - \frac{1}{I_x}K_d^R p + \frac{1}{I_x}LU_2 \tag{12}$$

$$q` = \frac{I_z - I_x}{I_y}pr - \frac{I_R}{I_y}p(w_1 + w_3 - w_2 - w_4) - \frac{1}{I_y}K_d^R q + \frac{1}{I_y}LU_3 \tag{13}$$

$$r` = \frac{I_x - I_y}{I_z}pq - \frac{1}{I_z}K_d^R r + \frac{1}{I_z}U_4 \tag{14}$$

$$\phi` = p + s_\phi s_\theta c_\theta^{-1} q + c_\phi s_\theta c_\theta^{-1} r \tag{15}$$

$$\theta` = c_\phi q - s_\phi r \tag{16}$$

$$\psi` = s_\phi c_\theta^{-1} q + c_\theta^{-1} c_\phi r \tag{17}$$

$$\mathbf{u} = \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{pmatrix} = M_{bd}\mathbf{w^2} = \begin{pmatrix} b & b & b & b \\ 0 & b & 0 & -b \\ b & 0 & -b & 0 \\ d & -d & d & -d \end{pmatrix} \begin{pmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{pmatrix} \tag{18}$$

It is worth noting that Equation (18) stands for an artificial input transformation made for ease of representation and control law definition. U_1 denotes the total lift force

produced by the rotors, U_2 and U_3 are the resultant forces generating pitch and roll movement respectively and U_4 is the resultant moment acting on the yaw axis. The presented quadrotor dynamical model has 12 state variables $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}) = (X, Y, Z, X', Y', Z', \phi, \theta, \psi, p, q, r)$ and 4 input variables $\mathbf{u} = (u_1, u_2, u_3, u_4) = (U_1, U_2, U_3, U_4)$. Note that the rotors' velocities giving the gyroscopic effect contribution still appear in Equations (12) and (13) but can be rewritten as a nonlinear function of the artificial input variables $f(\mathbf{u})$ by inversion and manipulation of Equation (18).

$$f(\mathbf{u}) = \sum_{i=1}^{4} (-1)^{i+1} \sqrt{|\,(M_{bd}{}^{-1}\mathbf{u})_i|} \;=\; w_1 + w_3 - w_2 - w_4 \tag{19}$$

The system, besides presenting under-actuation characteristics that always yield to challenging control design aspects, is highly nonlinear and coupled. When dealing with a dynamical system with such properties, it is convenient to derive a linearized model in order to obtain a clearer understanding of the relation between control and state variables and the system's stability. Furthermore, linearized models are often exploited for the design of the control law itself.

$$X`` = -\theta g \tag{20}$$

$$Y`` = -\phi g \tag{21}$$

$$Z`` = g - \frac{1}{m} U_1 \tag{22}$$

$$\phi`` = \frac{1}{I_x} L U_2 \tag{23}$$

$$\theta`` = \frac{1}{I_y} L U_3 \tag{24}$$

$$\psi`` = \frac{1}{I_z} U_4 \tag{25}$$

The equations from (20) to (25) refer to the linearization of the model represented by the equations from (9) to (17) by means of a Taylor expansion of the first order and adopting approximations for small angles, under the following assumptions:

- The quadrotor is hovering: external forces and moments are neglectable, linear and rotational velocities are small (the Euler rates coincide with the rotational velocities in the body frame).
- The equilibrium control input and state vectors are $(u_1^*, u_2^*, u_3^*, u_4^*) = (mg, 0,0,0)$ and $(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}) = (0,0,0,0,0,0,0,0,0,0,0,0)$, respectively.
- All terms of order higher than one are neglected.

2.2. System Control

If considering the linearized model in equations from (20) to (25) in the Laplace variable domain formulation, it is straightforward that the system is unstable due to the presence of poles in the origin. Additionally, the inputs U_2, U_3 and U_4 directly affect the quadrotor's attitude, while U_1 directly affects the altitude dynamics. Figure 3 shows the proposed control system's architecture, decoupled in two parts: the inner loop performs attitude stabilization, while the outer loop is responsible for altitude control and reference attitude computation, given a 3D trajectory to be followed. Each control input signal U_i is saturated before being given to the plant, i.e., the quadcopter. The lower and upper saturation bounds are U_i^{LB} and U_i^{UB}, respectively. The propellers' capacity is assumed to be such that the absolute value of the saturated total lift force U_1^{SAT} is equal to 3 mg. Analogously, an absolute value of 1.5 mg refers to U_2^{SAT} and U_3^{SAT}, while an absolute value equal to mgL refers to U_4^{SAT}.

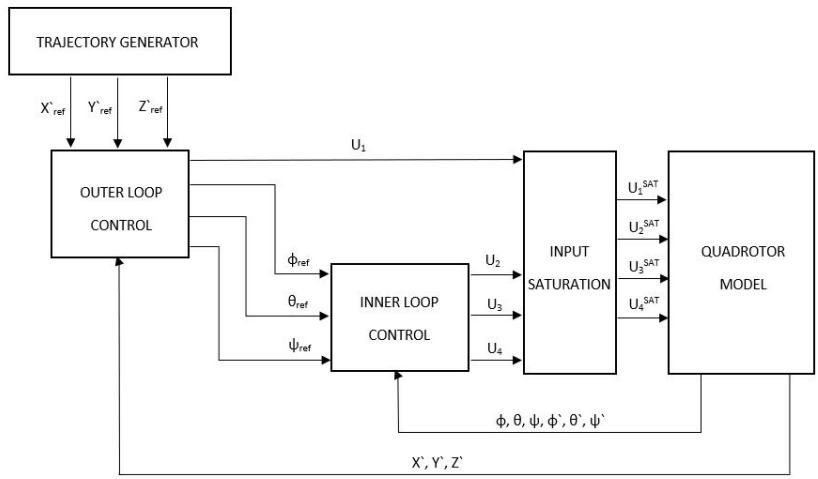

Figure 3. Overview of the proposed control system's architecture.

The inner loop is intended to be sufficiently faster than the outer loop so that the UAV's attitude can be effectively stabilized. Additionally, the state variables are assumed to be perfectly known to the controllers at each sampling instant. On the other hand, it is well known that the accuracy of position estimates may not be high enough to implement position-based trajectory controllers in many real-world applications. Consequently, the outer loop is intended to ensure tracking of velocity references.

As far as the inner loop is concerned, the choice of the best control law derives from a comparative evaluation of different linear and nonlinear controllers designed for attitude stabilization: PID, LQR, FL, SM and MPC. Both the tracking error and the controller command effort related to each controller are taken into account by means of the sum of the L_2 norms of state and controller command input signals. Each attitude stabilization controller is designed with simplified models but is applied to the complete nonlinear attitude system expressed by Equations (12) to (17) for the sake of the evaluation. Instead, the outer loop implements a dynamic inversion and PID-based velocity controller.

2.2.1. Inner Loop—PID Control

This follows the linearized attitude dynamical system expressed by equations from (23) to (25) in state space form, with $\mathbf{x} = (\phi, \theta, \psi, \phi', \theta', \psi')$ and $\mathbf{u} = (U_2, U_3, U_4)$. Three independent Proportional–Integral–Derivative controllers are designed with the aim of stabilizing the linearized system in hovering.

$$\begin{Bmatrix} \dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} \\ \mathbf{y} = C\mathbf{x} + D\mathbf{u} \end{Bmatrix} \tag{26}$$

$$A = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ LI_x^{-1} & 0 & 0 \\ 0 & LI_y^{-1} & 0 \\ 0 & 0 & I_z^{-1} \end{pmatrix}, C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, D = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

A classical formulation of the PID control law applied to the quadrotor stabilization problem in the time domain is presented in Equation (27). The three control parameters (K_p^i, K_i^i, K_d^i) of each control command are set in order to stabilize the model in (26) for the hovering equilibrium point. Refer to [25] for the theoretical foundations of PID control and further details.

$$u^p(t) = u^p_{eq} - K^p_p\left(y^p(t) - y^p_{eq}\right) - K^p_i \int_0^T \left(y^p(t) - y^p_{eq}\right)dt - K^p_d\left(\dot{y}^p(t) - \dot{y}^p_{eq}\right), u_{eq} = y_{eq} = (0,0,0), p = 1,2,3 \quad (27)$$

2.2.2. Inner Loop—LQR Control

The linearized attitude dynamical model in (26) is exploited to design the LQR-based attitude stabilizer. LQR controllers are based on optimization (minimization) of the sum of the weighted energies (L_2 norms) of the state and the control command, with the constraints that the system dynamics are respected. According to the Linear–Quadratic Regulator method, since the linearized system is completely controllable, i.e., the controllability matrix has full rank, it is possible to define the following feedback control action,

$$\mathbf{u}(t) = -K\mathbf{x}(t) \quad (28)$$

$$K = R^{-1}B^TP \quad (29)$$

$$A^TP + Q + PA - PBR^{-1}B^TP = 0 \quad (30)$$

$$Q = \begin{pmatrix} q_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & q_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & q_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & q_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & q_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & q_6 \end{pmatrix}, R = \begin{pmatrix} r_1 & 0 & 0 \\ 0 & r_2 & 0 \\ 0 & 0 & r_3 \end{pmatrix} \quad (31)$$

where K denotes the feedback control gain, Q and R are diagonal fixed weight matrices for the weighted L_2 norms of $\mathbf{x}$ and $\mathbf{u}$, respectively. The dynamics of the closed-loop system depend on the relative values of the control parameters: q_i, r_i. P can be computed by solving the Algebraic Riccati Equation formulated in Equation (30). Refer to [26] for the theoretical foundations of LQR control and further details.

2.2.3. Inner Loop—Feedback Linearization Control

This follows the attitude dynamical system exploited for the design of the feedback linearization-based attitude controller, with $\mathbf{x} = (\phi, \theta, \psi, \phi', \theta', \psi')$ and $\mathbf{u} = (U_2, U_3, U_4)$.

$$\dot{x}_p = x_{p+3}, p = 1,2,3 \quad (32)$$

$$\dot{x}_4 = \frac{I_Y - I_z}{I_x}x_5x_6 - \frac{I_R}{I_x}x_5f(\mathbf{u}) - \frac{1}{I_x}K^R_dx_4 + \frac{1}{I_x}Lu_1 \quad (33)$$

$$\dot{x}_5 = \frac{I_z - I_x}{I_y}x_4x_6 - \frac{I_R}{I_y}x_4f(\mathbf{u}) - \frac{1}{I_y}K^R_dx_5 + \frac{1}{I_y}Lu_2 \quad (34)$$

$$\dot{x}_6 = \frac{I_x - I_y}{I_z}x_4x_5 - \frac{1}{I_z}K^R_dx_6 + \frac{1}{I_z}u_3 \quad (35)$$

Coincidence between Euler rates and angular velocities is assumed for the attitude controller's design. The FL control law aims at linearizing the system with the term f^i_L and controlling it with a new control input term, u^*_i, which is intended as the sum of a proportional gain with respect to the Euler angles' errors and a derivative term, proportional to the Euler rates. Refer to [27] for the theoretical foundations of FL control and further details. Each control input is defined as follows.

$$u_1 = f^1_L + u^*_1 = \frac{I_x}{L}\left[-\frac{I_Y - I_z}{I_x}x_5x_6 + \frac{I_R}{I_x}x_5f(\mathbf{u}) + \frac{1}{I_x}K^R_dx_4 + D_1x_4 + P_1\left(x_{1_{ref}} - x_1\right)\right] \quad (36)$$

$$u_2 = f^2_L + u^*_2 = \frac{I_y}{L}\left[-\frac{I_z - I_x}{I_y}x_4x_6 + \frac{I_R}{I_y}x_4f(\mathbf{u}) + \frac{1}{I_y}K^R_dx_5 + D_2x_5 + P_2\left(x_{2_{ref}} - x_2\right)\right] \quad (37)$$

$$u_3 = f_L^3 + u_3^* = I_z\left[-\frac{I_x - I_y}{I_z}x_4x_5 + \frac{1}{I_z}K_d^R x_6 + D_3x_6 + P_3\left(x_{3_{ref}} - x_3\right)\right] \tag{38}$$

The closed-loop attitude dynamical subsystem becomes linear and decoupled, the dynamics of the system can be adjusted by tuning the pairs (D_i,P_i). This follows the closed-loop attitude dynamics of the controlled system.

$$\dot{x}_p = x_{p+3}\,, p = 1,2,3 \tag{39}$$

$$\dot{x}_4 = D_1x_4 + P_1\left(x_{1_{ref}} - x_1\right) \tag{40}$$

$$\dot{x}_5 = D_2x_5 + P_2\left(x_{2_{ref}} - x_2\right) \tag{41}$$

$$\dot{x}_5 = D_3x_6 + P_3\left(x_{3_{ref}} - x_3\right) \tag{42}$$

2.2.4. Inner Loop—Sliding Mode Control

The nonlinear attitude dynamical model in equations from (32) to (35) is exploited to design the sliding mode-based attitude stabilizer. The SM control method aims at the convergence of the state vector to another one, called the "sliding surface", which corresponds to an exponential achievement of the desired dynamical behavior of the system. The control input is the sum of two terms: a continuous term and a discontinuous one. The former is a continuous function of the state variables and the reference, u_c, that makes the "sliding surface", s, an invariant set. The discontinuous term, u_d, is added to achieve convergence to the "sliding surface". s is intended as a PD "sliding surface" with respect to the Euler angles and Euler rates tracking errors. Refer to [28,29] for the theoretical foundations of SM control and further details.

$$u_i = u_c^i + u_d^i\,, i = 1,2,3 \tag{43}$$

$$u_d^i = K_{di}\mathrm{sign}(s_i) = K_{di}\mathrm{sign}\left(x_{i+3_{ref}} - x_{i+3} + \beta_i\left(x_{i_{ref}} - x_i\right)\right) \sim K_{di}\frac{s_i}{\sqrt{s_i^2 + a}}\,, i = 1,2,3 \tag{44}$$

$$u_c^1 = \frac{I_x}{L}\left[-\frac{I_Y - I_z}{I_x}x_5x_6 + \frac{I_R}{I_x}x_5f(\mathbf{u}) + \frac{1}{I_x}K_d^R x_4 + \dot{x}_{4_{ref}} + \beta_1\left(x_{4_{ref}} - x_4\right)\right] \tag{45}$$

$$u_c^2 = \frac{I_y}{L}\left[-\frac{I_z - I_x}{I_y}x_4x_6 + \frac{I_R}{I_y}x_4f(\mathbf{u}) + \frac{1}{I_y}K_d^R x_5 + \dot{x}_{5_{ref}} + \beta_2\left(x_{5_{ref}} - x_5\right)\right] \tag{46}$$

$$u_c^3 = I_z\left[-\frac{I_x - I_y}{I_z}x_4x_5 + \frac{1}{I_z}K_d^R x_6 + \dot{x}_{6_{ref}} + \beta_3\left(x_{6_{ref}} - x_6\right)\right] \tag{47}$$

The $\mathrm{sign}\,(\cdot)$ function in the discontinuous part of the control action is approximated with a smooth function (with a small value of a) to avoid chattering, i.e., oscillations, when the state approaches the "sliding surface". The closed-loop attitude dynamics of the system can be adjusted by tuning the control parameters: a, K_{di}, β_i. This follows the closed-loop attitude dynamics of the controlled system.

$$\dot{x}_p = x_{p+3}\,, p = 1,2,3 \tag{48}$$

$$\dot{x}_4 = \dot{x}_{4_{ref}} + \beta_1\left(x_{4_{ref}} - x_4\right) + K_{d1}\frac{x_{4_{ref}} - x_4 + \beta_1\left(x_{4_{ref}} - x_4\right)}{\sqrt{\left(x_{4_{ref}} - x_4 + \beta_1\left(x_{4_{ref}} - x_4\right)\right)^2 + a}} \tag{49}$$

$$\dot{x}_5 = \dot{x}_{5_{ref}} + \beta_2\left(x_{5_{ref}} - x_5\right) + K_{d2}\frac{x_{5_{ref}} - x_5 + \beta_2\left(x_{5_{ref}} - x_5\right)}{\sqrt{\left(x_{5_{ref}} - x_5 + \beta_2\left(x_{5_{ref}} - x_5\right)\right)^2 + a}} \tag{50}$$

$$\dot{x}_6 = \dot{x}_{6_{ref}} + \beta_3\left(x_{6_{ref}} - x_6\right) + K_{d3}\frac{x_{6_{ref}} - x_6 + \beta_3\left(x_{6_{ref}} - x_6\right)}{\sqrt{\left(x_{6_{ref}} - x_6 + \beta_3\left(x_{6_{ref}} - x_6\right)\right)^2 + a}} \tag{51}$$

2.2.5. Inner Loop—Model Predictive Control

The state space form of the nonlinear attitude dynamical model in equations from (32) to (35) is exploited to design the model predictive attitude stabilizer. The MPC method is based on two phases: prediction and optimization. A prediction of the model's states, $\mathbf{x_p}$, is made over a certain prediction horizon, T_P, by integration of the model itself. An optimal control input $\mathbf{u}$ is chosen to bring the predicted state to the reference $\mathbf{r}$. The optimization problem formalized in (52) is solved to compute $\mathbf{u}^*(t \to t + T_P) = \left(U_2(t \to t + T_P), U_3(t \to t + T_P), U_4(t \to t + T_P)\right)$

$$\mathbf{u}^*(M) = \arg\min\ J(\mathbf{u}(M))\ \text{, subject to:} \tag{52}$$

$$J\left(\mathbf{u}(M)\right) = \int_t^{t+T_p}\left(\mathbf{x_{err}^T}(M)Q\mathbf{x_{err}}(M) + \mathbf{u^T}(M)R\mathbf{u}(M)\right)dM + \mathbf{x_{err}^T}(t + T_P)Q\mathbf{x_{err}}(t + T_P)\,,$$

$$\mathbf{x_{err}}(M) = \mathbf{r}(M) - \mathbf{x_p}(M)\,,$$

$$Q = \begin{pmatrix} q_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & q_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & q_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & q_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & q_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & q_6 \end{pmatrix}, R = \begin{pmatrix} r_1 & 0 & 0 \\ 0 & r_2 & 0 \\ 0 & 0 & r_3 \end{pmatrix}, P = \begin{pmatrix} p_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & p_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & p_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & p_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & p_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & p_6 \end{pmatrix},$$

$$\dot{\mathbf{x}}_\mathbf{P}(M) = f\left(\mathbf{x_p}(M), \mathbf{u}(M)\right)\ ,\ \mathbf{x_p}(M) = \mathbf{x}(M)\,,$$

$$\mathbf{u}(M) \in [\mathbf{U_{LB}}, \mathbf{U_{UP}}]\,, M = [t \to t + T_P]$$

where $\mathbf{U_{LB}}$ and $\mathbf{U_{UP}}$ denote the lower and upper bounds of the optimal control input $\mathbf{u}^*$, which would be subject to saturation anyway, $\mathbf{x_{err}}$ is the predicted state's error vector, Q, R and P denote diagonal matrices that contain the cost function $J\left(\mathbf{u}(M)\right)$ weight factors. The receding horizon strategy is applied to the aforementioned formulation of the problem at each time T_z. T_s denotes the sampling time. A constant input parametrization is chosen such that $\mathbf{u}^*(t \to t + T_z) = \mathbf{u}^*(T_z)$. q_i, r_i, p_i, T_s, T_p and T_z represent the control parameters. Refer to [30] for the theoretical foundations of MPC and further details.

2.2.6. Outer Loop

A simplified translational dynamical model is reported in state space form, with $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5, x_6) = (X', Y', Z', \phi, \theta, \psi)$ and $u = U_1$.

$$\dot{x}_1 = -\frac{1}{m}\left(s_{x_6}s_{x_4} + c_{x_6}s_{x_5}c_{x_4}\right)u \tag{53}$$

$$\dot{x}_2 = -\frac{1}{m}\left(s_{x_5}s_{x_6}c_{x_4} - s_{x_4}c_{x_6}\right)u \tag{54}$$

$$\dot{x}_3 = g - \frac{1}{m}c_{x_5}c_{x_4}u \tag{55}$$

The aim of the outer loop controller is to control the quadcopter's altitude and to compute the reference Euler angles to be tracked by the inner loop controller. The outer loop controller is divided into two parts: the first part implements three PID controllers

that compute the reference linear accelerations from the linear velocity errors, the second part performs a dynamics inversion of the system in equations from (53) to (55), with the assumption that $\psi_{ref} = 0$, for the computation of the reference pitch and roll angles, θ_{ref} and ϕ_{ref}, respectively, and the total lift force, u.

$$\dot{x}_{ref}{}^{p}(t) = -K_p^p\left(x^p(t) - x_{ref}^p(t)\right) - K_i^p \int_0^T \left(x^p(t) - x_{ref}^p(t)\right)dt - K_d^p\left(x^p(t) - x_{ref}^p(t)\right), p = 1,2,3 \tag{56}$$

$$\beta = -\mathrm{sign}\left(\dot{x}_{ref}{}^{1}\right)\left(1 + \left(\frac{g - \dot{x}_{ref}{}^{3}}{\dot{x}_{ref}{}^{1}}\right)^2\right)^{-\frac{1}{2}} \tag{57}$$

$$x_{ref}^4 = \arcsin\left(\frac{Bm}{u}\right) \tag{58}$$

$$x_{ref}^5 = \arcsin(\beta) \tag{59}$$

$$x_{ref}^6 = 0 \tag{60}$$

$$u = m\sqrt{\frac{(\dot{x}_{ref}{}^{1})^2}{\beta^2} + (\dot{x}_{ref}{}^{2})^2} \tag{61}$$

Equations from (56) to (61) express the full outer loop trajectory controller. The control parameters (K_p^i, K_i^i, K_d^i) can be set in order to ensure satisfactory dynamics for different 3D complex trajectories of reference, expressed by the signals $x_{ref}^i(t)$. In order to avoid numerical problems, $\dot{x}_{ref}{}^{1}$ is forced to have a lower bound different from zero in absolute value. A similar approach is proposed in [11].

2.2.7. Trajectory Generator

The ascent helix and the complex ascent helix are examples of relatively hard trajectories for quadcopters. They are considered as a benchmark for demonstrating the complete control system's capability of executing complex trajectories. The ascent helix and the complex ascent helix are expressed as follows,

$$\dot{X}_{ref}{}^{1}(t) = -f_x^1 \sin(f_x^1 t)\ , \dot{X}_{ref}{}^{2}(t) = -f_x^2 \sin(f_x^2 t) - 2\, f_x^2 \sin(f_x^2 t)\cos(f_x^2 t) \tag{62}$$

$$\dot{Y}_{ref}{}^{1}(t) = f_y^1 \cos\left(f_y^1 t\right)\ , \dot{Y}_{ref}{}^{2}(t) = f_y^2 \cos\left(f_y^2 t\right) + 2\, f_y^2 \cos\left(f_y^2 t\right)\sin\left(f_y^2 t\right) \tag{63}$$

$$\dot{Z}_{ref}{}^{1}(t) = \dot{Z}_{ref}{}^{2}(t) = a \tag{64}$$

where superscripts 1 and 2 indicates the ascent helix and complex ascent helix trajectories, respectively. t is the simulation time, f_x, f_y denotes the frequency of the signal components on the plane OXY and a denotes the climb velocity. The equations from (62) to (64) represent the derivation with respect to t of reference trajectories designed in space and time, expressed by the position variables $X_{ref}(t)$, $Y_{ref}(t)$, $Z_{ref}(t)$.

2.3. Simulation Setting

The whole control architecture is implemented in the MATLAB/Simulink® software environment, mainly by means of MATLAB function blocks. Table 1 shows the simulation parameters, while the control parameter settings of all the proposed attitude stabilizers are reported in Table 2. All the values are expressed according to the International System of Units.

Table 1. Simulation parameters.

Simulation Parameter Type	Parameter	Value
UAV	m	0.60 kg
	g	9.81 $m \cdot s^{-2}$
	b	$3.60 \cdot 10^{-6}$ $kg \cdot m$
	d	$1.60 \cdot 10^{-7}$ $kg \cdot m^2$
	I_x	$5.20 \cdot 10^{-3}$ $kg \cdot m^2$
	I_y	$5.20 \cdot 10^{-3}$ $kg \cdot m^2$
	I_z	$9.40 \cdot 10^{-3}$ $kg \cdot m^2$
	I_R	$4.20 \cdot 10^{-5}$ $kg \cdot m^2$
	L	0.20 m
	K_d^R	0.15 $kg \cdot m^2 \cdot s^{-1}$
	K_d^L	0.15 $kg \cdot s^{-1}$
Saturation	$[U_1^{LB}, U_1^{UB}]$	$[-17, 17]$ N
	$[U_2^{LB}, U_2^{UB}]$	$[-9, 9]$ N
	$[U_3^{LB}, U_3^{UB}]$	$[-9, 9]$ N
	$[U_4^{LB}, U_4^{UB}]$	$[-1.2, 1.2]$ $N \cdot m$
Trajectory Generator	$f_x^1 = f_y^1$	1 s^{-1}
	a	-1 $m \cdot s^{-1}$
	$f_x^2 = f_y^2$	1 s^{-1}
Outer Loop Controller	$K_p^1 = K_p^2 = K_p^3$	8
	$K_i^1 = K_i^2 = K_i^3$	8.5
	$K_d^1 = K_d^2 = K_d^3$	0.01
	Derivative Filter Coefficient	100

Table 2. Attitude stabilization controllers' parameters.

Inner Loop Controller	Control Parameter	Value
PID	$K_p^1 = K_p^2 = K_p^3$	0.56
	$K_i^1 = K_i^2 = K_i^3$	0.28
	$K_d^1 = K_d^2 = K_d^3$	0.23
	Derivative Filter Coefficient	49.74
LQR	$[q_1, q_2, q_3, q_4, q_5, q_6]$	[0,0,0,1,1,1]
	$r_1 = r_2 = r_3$	0.0001
FL	$[D_1, P_1]$	$[-120,38]$
	$[D_2, P_2]$	$[-120,38]$
	$[D_3, P_3]$	$[-50,20]$
SM	$[K_{d1}, \beta_1]$	[0.2,20]
	$[K_{d2}, \beta_2]$	[0.2,20]
	$[K_{d3}, \beta_3]$	[0.05,5]
	a	0.1
MPC	T_s	0.01 s
	T_p	0.1 s
	T_z	0.05 s
	$[q_1, q_2, q_3, q_4, q_5, q_6]$	[10,10,1,0,0,0]
	$r_1 = r_2 = r_3$	0.0001
	$[p_1, p_2, p_3, p_4, p_5, p_6]$	[1,1,1,0,0,0]

3. Simulation Results

3.1. Inner Loop

All the attitude stabilizers proposed in Sections 2.2.1–2.2.5 are shown to be able to stabilize the nonlinear quadrotor attitude model, expressed by Equations (12)–(17), yet at different performance levels. All the simulations are performed considering the same initial condition for the plant, with the quadrotor having initial roll, pitch and yaw angles of 20° and initial roll, pitch and yaw rates of 10 rad/s. The gyroscopic effect is part of the nonlinear attitude dynamical system and is computed assuming a constant total lift force equal to the quadcopter's weight in Equation (19). Figures 4–8 show the Euler angles signals of the PID, LQR, FL, SMC and MPC-based attitude stabilizers, respectively.

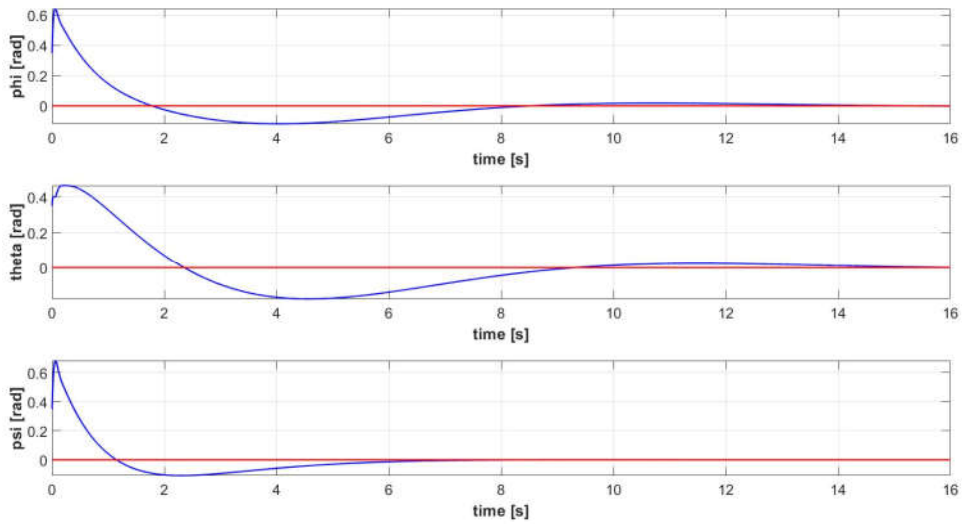

Figure 4. Euler angles signals $[\phi(t), \vartheta(t), \psi(t)]$ of the controlled attitude system (blue): PID-based Attitude Stabilizer (red).

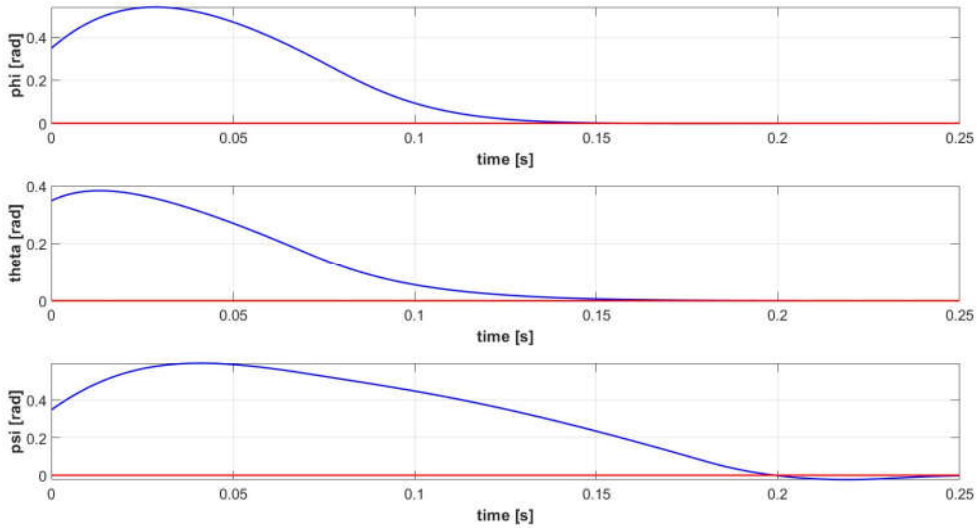

Figure 5. Euler angles signals $[\phi(t), \vartheta(t), \psi(t)]$ of the controlled attitude system (blue): LQR-based Attitude Stabilizer (red).

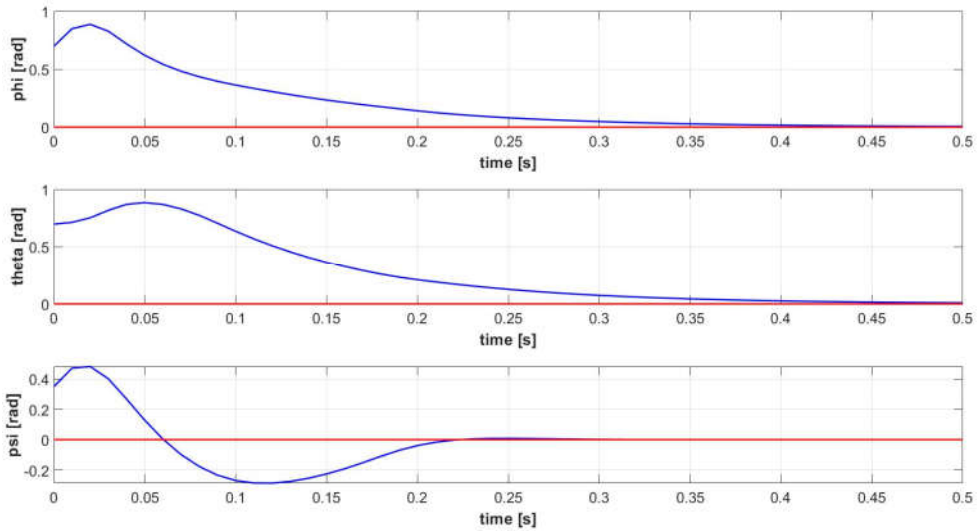

Figure 6. Euler angles signals $[\phi(t), \vartheta(t), \psi(t)]$ of the controlled attitude system (blue): FL-based Attitude Stabilizer (red).

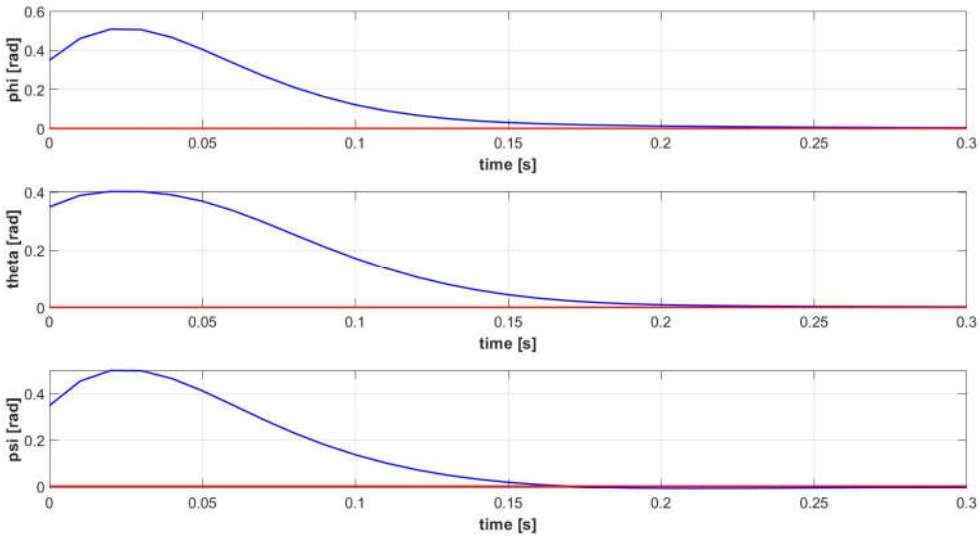

Figure 7. Euler angles signals $[\phi(t), \vartheta(t), \psi(t)]$ of the controlled attitude system (blue): SMC-based Attitude Stabilizer (red).

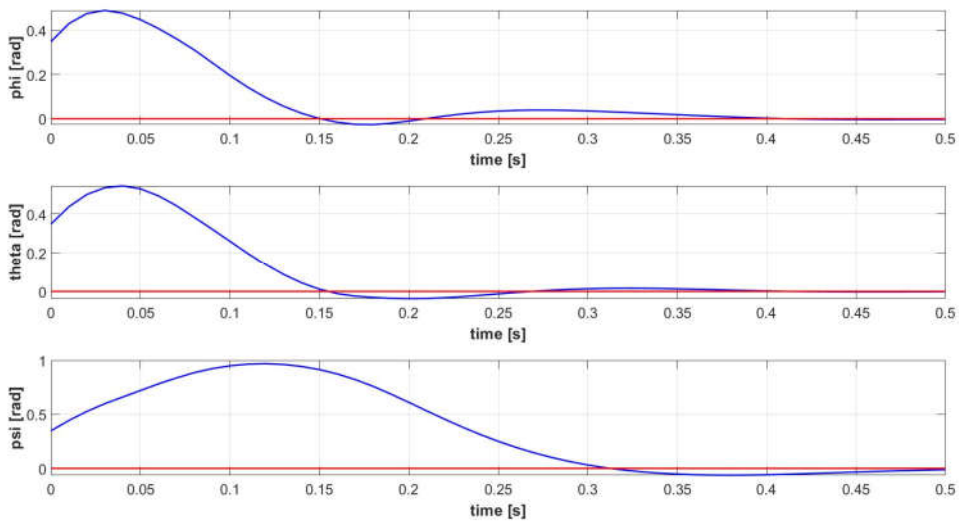

Figure 8. Euler angles signals $[\phi(t), \vartheta(t), \psi(t)]$ of the controlled attitude system (blue): MPC-based Attitude Stabilizer (red).

The three pairs of proportional, integral and derivative gains of the PID-based attitude controller are tuned exploiting the automatic tuning library in MATLAB applied to the linearized system in (26), considering the control input saturation, to achieve a fast (rise time of ~1 s) and robust step response. As shown in Figure 4, the Euler angles dynamics relative to the PID-based stabilizer are much slower with respect to all the others, meaning that PID control is effective but requires extensive gain tuning and is not robust to model uncertainties, as the performances significantly decrease when the controller is applied to the nonlinear attitude system. As far as the classical PID control approach is concerned, which is the approach adopted in this work, a systematic procedure exists for tuning the controller's gains; however, a highly time-consuming offline gains optimization is required when dealing with nonlinear systems, as for quadrotor UAVs. The amount of computation related to the online optimization of the MPC-based attitude stabilizer is such that the optimal solution is computed, on average, within 0.01 s. A control horizon $T_z = 0.05s$ is chosen in order to guarantee the optimal controller command at each step. A trial-and-error tuning procedure is adopted for the choice of the control parameters of the other attitude stabilizers, which show faster dynamics and greater robustness to model uncertainties. The sum of the L_2 norms of the Euler angles' tracking error signals, L_2^x, is chosen as the approximative evaluation parameter of the controllers' tracking performance. Analogously, the sum of the L_2 norms of the controller command signals, L_2^u, represents the approximative evaluation parameter of the expected command activity of an on-board hardware implementation of the controller. Since the L_2 norm of a signal represents the "energy" associated with the signal itself, it can be considered as an approximative quantitative estimation of both the overall tracking error and the controller command effort. Table 3 shows the controllers' evaluation parameters referring to the attitude stabilization simulations presented in this section.

Table 3. Evaluation parameters: sum of L_2 norms of the controller commands signals, L_2^u, and sum of L_2 norms of the Euler angles error signals, L_2^x.

Attitude Stabilizer.	L_2^u	L_2^x
PID	1.60	1.40
LQR	4.53	0.41
FL	5.6	0.64
SMC	4.35	0.36

MPC	3.70	0.65

The proposed attitude stabilizers are ranked from best to worse as SMC, LQR, FL, MPC, PID for L_2^x and as PID, MPC, SMC, LQR, FL for L_2^u. Considering the attitude controllers that lead to the fastest dynamics (SMC, LQR and FL), the choice of the inner loop control law derives from a trade-off between tracking performances and controller command effort, since the battery consumption, being related to the forces and moments required to act on the quadcopter platform, is a key limitation factor when dealing with multirotor UAVs executing medium-to-long range missions. The SMC-based attitude stabilizer is the most promising one, having minimum L_2^x, intermediate L_2^u and ensuring fast dynamics to the controlled variables. Besides the norm evaluation, the FL-based controller yields to bigger peaks in the controlled signals. It is worth noting that ad-hoc simulation campaigns may be designed in order to tune the control parameters of each attitude stabilizer such that the system achieves a reference dynamic response. This could enhance performance, but the inherent features of each control approach would not change, as well as the highlighted differences among the methods.

3.2. Closed-Loop System

The complete controlled quadrotor system is proven to be able to execute complex 3D trajectories, defined by Equations (62)–(64). Both an ascent helix (trajectory 1) and a more complex ascent helix (trajectory 2) are simulated. The signals related to the controlled variables are shown in Figure 9 for trajectory 1, Figure 10 shows the corresponding quadcopter's 3D trajectory, Figure 11 shows the control command signals required for executing trajectory 1. The Euler Angles signals related to trajectory 1 are shown in Figure 12. The signals related to the controlled variables are shown in Figure 13 for trajectory 2, Figure 14 shows the corresponding quadcopter's 3D trajectory, Figure 15 shows the control command signals required for executing trajectory 2. The Euler Angles signals related to trajectory 2 are shown in Figure 16. Considering that either PID control is a model-free approach with an easy gain tuning procedure, or the outer loop controller is intended to ensure a much slower dynamic response with respect to the inner loop controller (order of a few seconds), a PID controller in the outer loop is sufficient to ensure a satisfactory dynamic response. The tracking performances are satisfactory for either the sinusoidal-based or the unitary step reference velocity signals, as shown in Figures 9 and 13, which spatially derive in the helicoidal maneuvers shown in Figures 10 and 14. Due to the inherent characteristics of PID control, a relatively small delay between the sinusoidal-based reference velocity signal and the actual quadcopter velocity is inevitable. That is reflected also in a smaller, yet present, delay in the tracking of the consequent sinusoidal-based Euler angles signals, as shown in Figures 12 and 16. Overall, the proposed architecture can effectively control the quadcopter during complex maneuvers without exceeding the saturation limits, as shown by the signals of the controller commands in Figures 11 and 15.

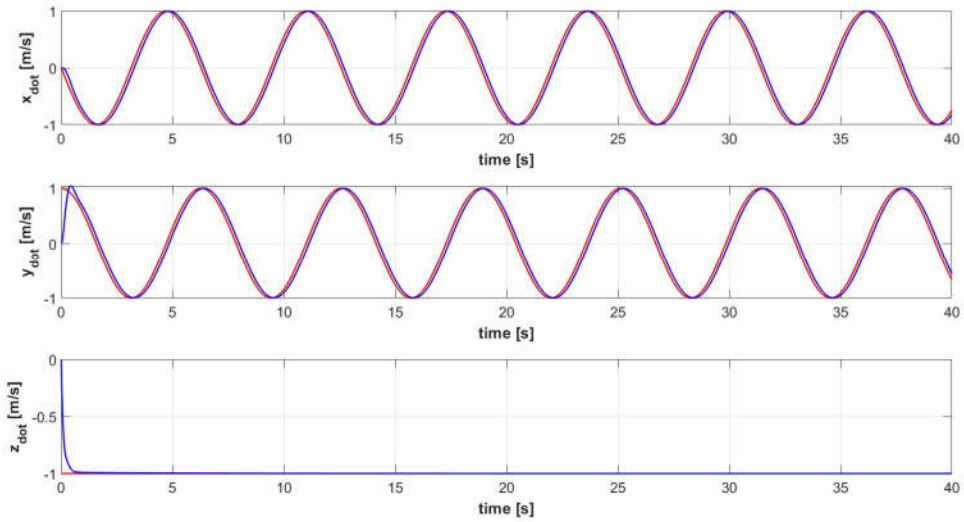

Figure 9. Velocity signals [X`(t), Y`(t), Z`(t)] of the controlled quadrotor system (blue) and reference velocity signals of trajectory 1 (red).

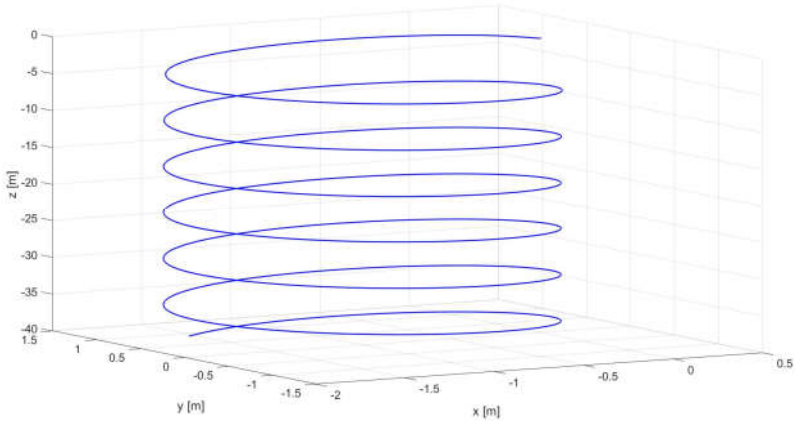

Figure 10. 3D view of trajectory 1: position trajectory of the controlled quadrotor system.

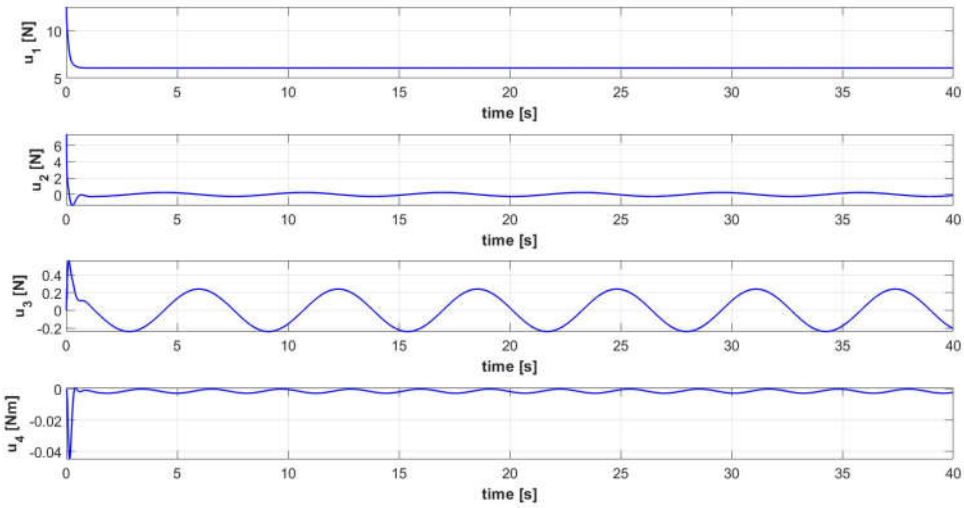

Figure 11. Control input signals $[u_1(t), u_2(t), u_3(t), u_4(t)]$ of the controlled quadrotor system executing trajectory 1.

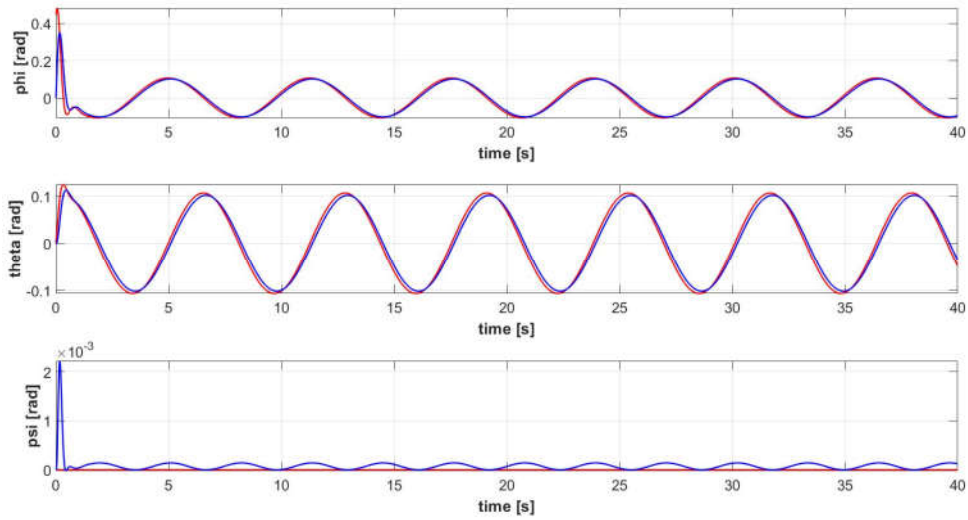

Figure 12. Euler angles signals $[\phi(t), \vartheta(t), \psi(t)]$ of the controlled quadrotor system (blue) and reference attitude signals for trajectory 1 (red).

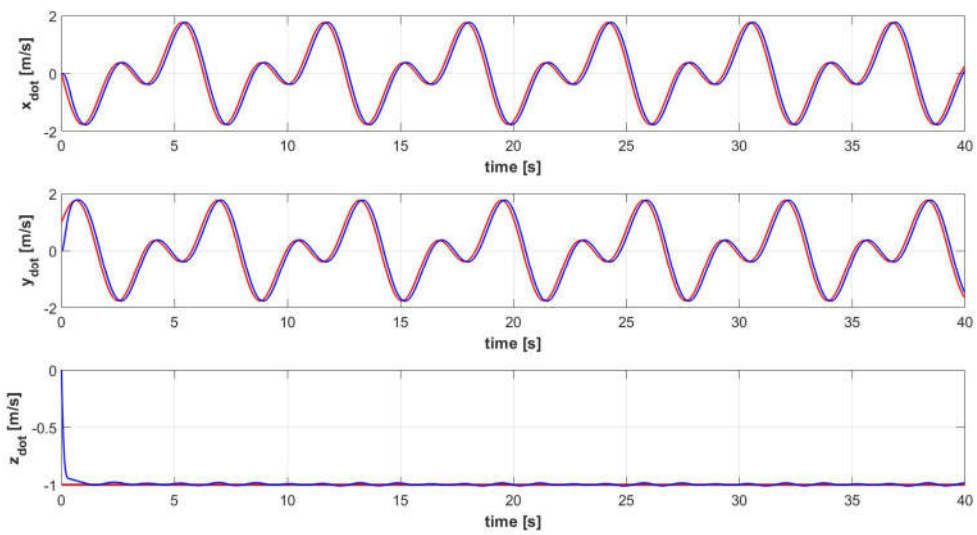

Figure 13. Velocity signals $[X`(t), Y`(t), Z`(t)]$ of the controlled quadrotor system (blue) and reference velocity signals of trajectory 2 (red).

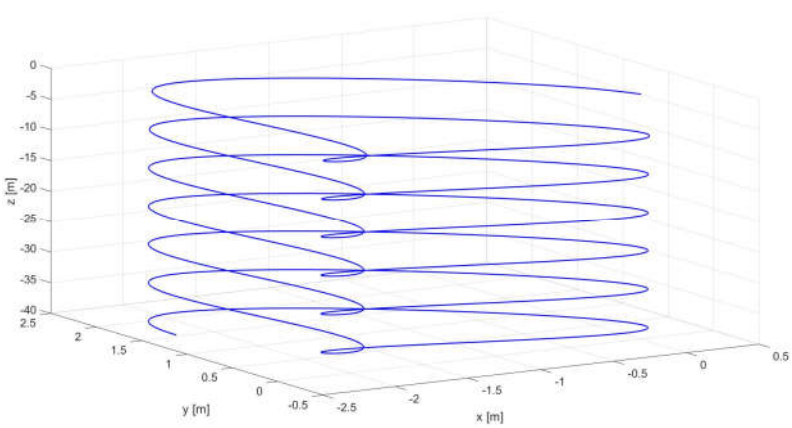

Figure 14. 3D view of trajectory 2: position trajectory of the controlled quadrotor system.

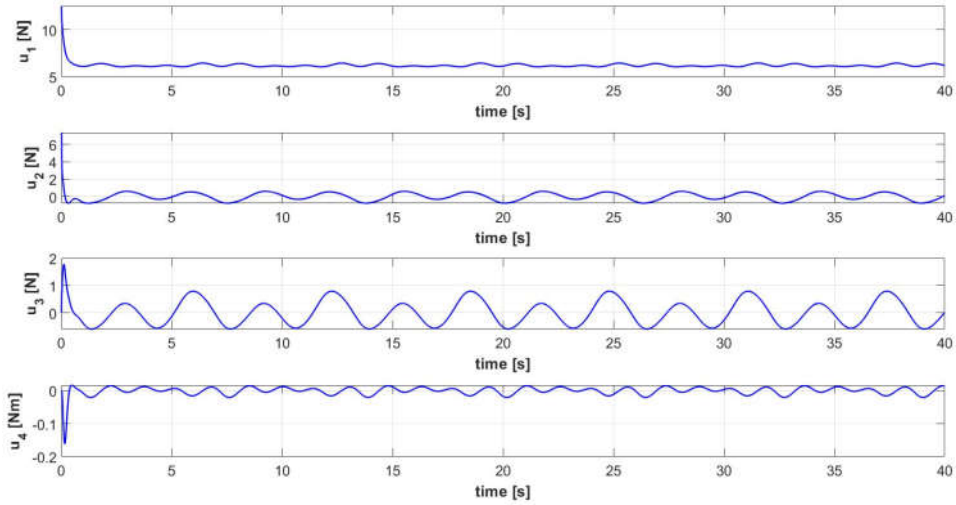

Figure 15. Control input signals $[u_1(t), u_2(t), u_3(t), u_4(t)]$ of the controlled quadrotor system executing trajectory 2.

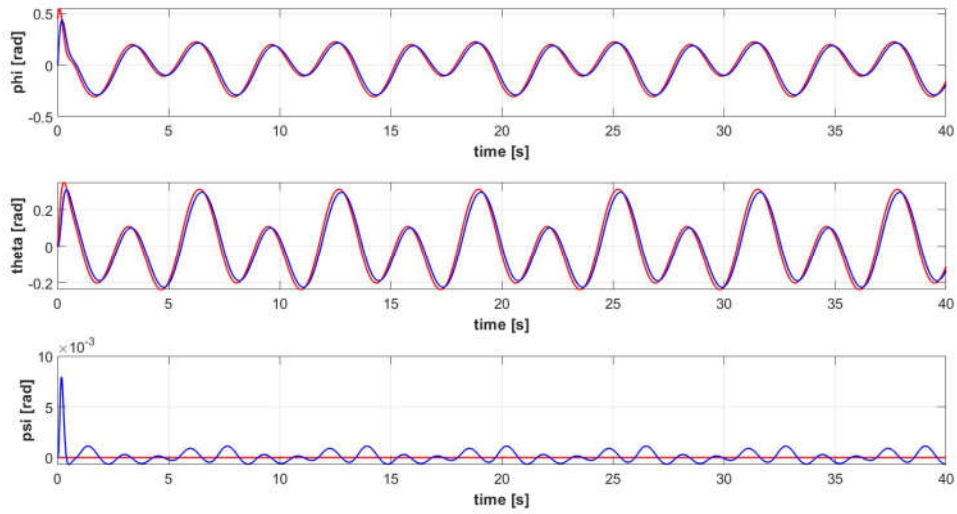

Figure 16. Euler angles signals $[\phi(t), \vartheta(t), \psi(t)]$ of the controlled quadrotor system (blue) and reference attitude signals for trajectory 2 (red).

4. Conclusions

In this paper, a control system architecture for 3D trajectory tracking is implemented and simulated in MATLAB/Simulink® for a micro-quadrotor UAV. The highly nonlinear, coupled, unstable and underactuated plant system, i.e., the quadrotor, is modelled with the Newton–Euler formalism. The control architecture is intended to stabilize the quadrotor attitude dynamics, which is fully actuated with a sufficiently fast inner loop controller, and to track velocity references in the outer loop. A comparative study of well-known linear and nonlinear model-free and model-based control laws applied to the attitude stabilization problem is proposed: PID, LQR, FL, SMC and MPC attitude stabilization laws are designed and implemented. The choice of the inner loop controller derives from either the fast dynamics requirement of the inner loop or an L_2 norm-based approximative evaluation of both tracking error and controller command signals since the relatively poor battery autonomy is still one of the main limitations for the deployment of multirotor UAVs in medium-to-long range missions. The reference attitude signals for the inner loop controller are derived from a simplified translational dynamics inversion

performed in an outer loop controller, which is also responsible for altitude control. In the outer loop, the dynamics inversion is combined with three independent PID controllers, while the Sliding Mode attitude stabilizer is implemented in the inner loop. A velocity trajectory generator and a control input saturation block complete the proposed control architecture. The controlled closed-loop quadrotor system is proven to be able to execute complex 3D trajectories, such as ascent helixes. Consistent with the aim of this work, the main summarizing considerations are reported in Table 4 in order to highlight the main observations drawn from the study and provide guidelines for further research in this field.

Future research directions will include the realization of other comparative studies for control of quadrotor UAVs, considering external disturbances, e.g., the wind, and parametric uncertainties, e.g., variable payloads.

Table 4. Summarizing table of considerations drawn from the study. The number of diamonds is proportional to the "suitability of the method to UAV control".

Control Method	Advantages	Disadvantages	Overall Suitability to UAV Control
PID	• Systematic gain tuning method for linear systems • Possible zero steady state error • Model free	• No constraints handling • Poor robustness • No multiple inputs and multiple outputs handling • Extensive offline gain optimization	◊
LQR	• Possible handling of multiple inputs and multiple outputs • Possible handling of performance and command effort • Fast dynamics	• May fail to overcome steady-state error	◊◊◊
FL	• Systematic cancellation of nonlinearities	• No constraints handling • Poor robustness • Model in affine form only	◊◊
SMC	• Robustness • Fast dynamics • Good performances with highly nonlinear systems	• Chattering may cause instability	◊◊◊◊
MPC	• Possible handling of multiple inputs and multiple outputs • Possible handling of performance and command effort • Possible handling of input and output constraints	• Slow dynamics	◊◊

Author Contributions: All authors contributed extensively to the current study. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used in the current study are available from the corresponding author upon reasonable request.

Acknowledgments: The authors wish to thank the Piedmont Aerospace District for supporting the PhD activity of Marco Rinaldi.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cohen, A.; Shaheen, A.; Farrar, E. Urban Air Mobility: History, Ecosystem, Market Potential, and Challenges. *IEEE Trans. Intell. Transp. Syst.* **2022**, *22*, 6074–6087.
2. Nawaz, H.; Ali, H.M.; Massan, S.U.R. Applications of unmanned aerial vehicles: A review. *3c Tecnol. Glosas De Innovación Apl. A La Pyme. Special Issue* **2019**, *2019*, 85–105.
3. Aslan, M.F.; Durdu, A.; Sabanci, K.; Ropelewska, E.; Gültekin, S.S. A Comprehensive Survey of the Recent Studies with UAV for Precision Agriculture in Open Fields and Greenhouses. *Appl. Sci.* **2022**, *12*, 1047.
4. Skrinjar, J.P.; Skorput, P.; Furdic, M. Application of Unmanned Aerial Vehicles in Logistic Processes. In Proceedings of the New Technologies, Development and Applications (NT-2018), Sarajevo, Bosnia-Herzegovina, 14–16 June 2018.
5. Idrissi, M.; Salami, M.R.; Annaz, F. A Review of Quadrotor Unmanned Aerial Vehicles: Applications, Architectural Design and Control Algorithms. *J. Intell. Robot. Syst.* **2022**, *104*, 22.
6. Roy, R.; Islam, M.; Sadman, N.; Mahmud, M.A.P.; Gupta, K.D.; Ahsan, M.M. A Review on Comparative Remarks, Performance Evaluation and Improvement Strategies of Quadrotor Controllers. *Technologies* **2021**, *9*, 37.
7. Shauqee, M.N.; Rajendran, P.; Suhadis, N.M. Quadrotor Controller Design Techniques and Applications Review. *Incas Bull.* **2021**, *13*, 179–194.
8. Zulu, A.; John, S. A Review of Control Algorithms for Autonomous Quadrotors. *Open J. Appl. Sci.* **2014**, *4*, 547–556.
9. Shraim, H.; Awada, A.; Youness, R. A survey on quadrotors: Configurations, modeling and identification, control, collision avoidance, fault diagnosis and tolerant control. *IEEE Aerosp. Electron. Syst. Mag.* **2018**, *33*, 14–33.
10. Bolandi, H.; Rezaei, M.R.; Mohsenipour, R.; Nemati, H.; Smailzadeh, S.M. Attitude Control of a Quadrotor with Optimized PID Controller. *Intell. Control Autom.* **2013**, *4*, 335–342.
11. Voos, H.; Nonlinear control of a quadrotor micro-UAV using feedback-linearization. In Proceedings of the 2009 IEEE International Conference on Mechatronics, Malaga, Spain, 14–17 April 2009.
12. Islam, M.; Okasha, M.; Idres, M. Dynamics and Control of Quadcopter using Linear Model Predictive Control Approach. In Proceedings of the AEROS Conference 2017, Putrajaya, Malaysia, 12 December 2017.
13. Madani, T.; Benallegue, A. Backstepping Control for a Quadrotor Helicopter. In Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, 9–15 October 2006.
14. Petrusevski, I.; Aleksandar, R. Simple fuzzy solution for quadrotor attitude control. In Proceedings of the 12th Symposium on Neural Network Applications in Electrical Engineering (NEUREL), Belgrade, Serbia, 25–27 November 2014.
15. Capello, E.; Guglieri, G.; Quagliotti, F.; Sartori, D. Design and Validation of an L1 Adaptive Controller for Mini-UAV Autopilot. *J. Intell. Robot. Syst.* **2013**, *69*, 109–118.
16. Noordin, A.; Mohd Basri, M.A.; Mohamed, Z. Position and Attitude Tracking of MAV Quadrotor Using SMC-Based Adaptive PID Controller. *Drones* **2022**, *6*, 263.
17. Sankaranarayanan, V.N.; Satpute, S.; Nikolakopoulos, G. Adaptive Robust Control for Quadrotors with Unknown Time-Varying Delays and Uncertainties in Dynamics. *Drones* **2022**, *6*, 220.
18. Shakeel, T.; Arshad, J.; Jaffery, M.H.; Rehman, A.U.; Eldin, E.T.; Ghamry, N.A.; Shafiq, M. A Comparative Study of Control Methods for X3D Quadrotor Feedback Trajectory Control. *Appl. Sci.* **2022**, *12*, 9254.
19. Okasha, M.; Kralev, J.; Islam, M. Design and Experimental Comparison of PID, LQR and MPC Stabilizing Controllers for Parrot Mambo Mini-Drone. *Aerospace* **2022**, *9*, 298.
20. Available online: https://basecamp-shop.com/en/product/dji-quadcopter-flame-wheel-arf-kit-450 (accessed on 18 November 2022).
21. Ghazbi, S.N.; Aghli, Y.; Alimohammadi, M.; Akbari, A.A. Quadrotors Unmanned Aerial Vehicles: A Review. *Int. J. Smart Sens. Intell. Syst.* **2016**, *9*, 309–333.
22. Zhang, X.; Li, X.; Wang, K.; Lu, Y. A Survey of Modelling and Identification of Quadrotor Robot. *Abstr. Appl. Anal.* **2014**, *2014*, 1–16.
23. Moness, M.; Bakr, M. Development and Analysis of Linear Model Representations of the Quad-Rotor System. In Proceedings of the Aerospace Sciences and Aviation Technology, Cairo, Egypt, 26–28 May 2015.
24. Naidoo, Y.; Stopforth, R.; Bright, G. Quad-Rotor Unmanned Aerial Vehicle Helicopter Modelling & Control. *Int. J. Adv. Robot. Syst.* **2011**, *8*, 139–149.
25. Astrom, K.J.; Hagglund, T. *Advanced PID Control*; Instrumentation, Systems and Automation Society: Pittsburgh, PA, USA, 2006.
26. Dorato, P.; Abdallah, C.; Cerone, V. *Linear Quadratic Control: An Introduction*; Krieger Pub Co.: Malabar, FL, USA, 2000.
27. Krener, A.J. Feedback Linearization. In *Mathematical Control Theory*; Baillieul, J., Willems, J.C., Eds.; Springer: New York, NY, USA, 1999; pp. 66–98.
28. Edwards, C.; Spurgeon, S. *Sliding Mode Control: Theory and Applications*; CRC Press: London, UK, 1998.

29. Yan, B.; Dai, P.; Liu, R.; Xing, M.; Liu, S. Adaptive super-twisting sliding mode control of variable sweep morphing aircraft. *Aerosp. Sci. Technol.* **2019**, *92*, 198–210.
30. Rawlings, J.B.; Mayne, D.Q.; Diehl, M.M. *Model Predictive Control: Theory, Computation, and Design*; Nob Hill Publishing: Santa Barbara, CA, USA, 2020.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.