Article

Improving 3D Path Tracking of Unmanned Aerial Vehicles through Optimization of Compensated PD and PID Controllers

Nadia Samantha Zuñiga-Peña [†] , Norberto Hernández-Romero ^{*,†} , Juan Carlos Seck-Tuoh-Mora [†] ,
Joselito Medina-Marin [†] and Irving Barragan-Vite [†]

Área Académica de Ingeniería, Instituto de Ciencias Básicas e Ingeniería, Universidad Autónoma del Estado de Hidalgo, Carr. Pachuca-Tulancingo km. 4.5, Pachuca 42184, Hidalgo, Mexico; zu450520@uaeh.edu.mx (N.S.Z.-P.); jseck@uaeh.edu.mx (J.C.S.-T.-M.); jmedina@uaeh.edu.mx (J.M.-M.); irvingb@uaeh.edu.mx (I.B.-V.)

* Correspondence: nhromero@uaeh.edu.mx; Tel.: +52-7711275652

† These authors contributed equally to this work.

Abstract: The development of quadrotor unmanned aerial vehicles (QUAVs) is a growing field due to their wide range of applications. QUAVs are complex nonlinear systems with a chaotic nature that require a controller with extended dynamics. PD and PID controllers can be successfully applied when the parameters are accurate. However, this parameterization process is complicated and time-consuming; most of the time, parameters are chosen by trial and error without guaranteeing good performance. The originality of this work is to present a novel nonlinear mathematical model with aerodynamic moments and forces in the Newton–Euler formulation, and identify metaheuristic algorithms applied to parameter optimization of compensated PD and PID controls for tracking the trajectories of a QUAV. Eight metaheuristic algorithms (PSO, GWO, HGS, LSHADE, LSPACMA, MPA, SMA and WOA) are reported, and RMSE is used to measure each dynamic performance of the simulations. For the PD control, the best performance is obtained with the HGS algorithm with an RMSE = 0.037247252379126. For the PID control, the best performance is obtained with the HGS algorithm with an RMSE = 0.032594309723623. Trajectory tracking was successful for the QUAV by minimizing the error between the desired and actual dynamics.

Keywords: quadrotor unmanned aerial vehicles; compensated PD controller; compensated PID controller; metaheuristics algorithms; Newton-Euler formulation; nonlinear system; parameterization

Citation: Zuñiga-Peña, N.S.; Hernández-Romero, N.; Seck-Tuoh-Mora, J.C.; Medina-Marin, J.; Barragan-Vite, I. Improving 3D Path Tracking of Unmanned Aerial Vehicles through Optimization of Compensated PD and PID Controllers. *Appl. Sci.* **2022**, *12*, 99. <https://doi.org/10.3390/app12010099>

Academic Editor: Seong-Ik Han

Received: 3 November 2021

Accepted: 15 December 2021

Published: 23 December 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 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

One of the most complex engineering problems is the limitation of algorithms to control unmanned aerial vehicles (UAVs). UAVs are usually referred to as drones, and are devices that can be autonomous or remotely piloted. They can also be multirotor, a manoeuvrability characteristic that enables vertical take-off and landing. Since 1970, study and development of QUAVs have been an active research topic for governments, industry, and scientists around the world [1].

Quadrotor UAV (QUAV) is the most simple UAV and contains four motors. QUAVs have a broad variety of applications, i.e., supervision and inspection, monitoring, imaging and mapping, search and rescue operations, and load carrying [2]. QUAVs complexity is mostly related to the fact that QUAVs are underactuated and nonlinear systems that respond in coupled dynamics and chaotic behaviors [3]. These features make them challenging to model and improve the design of controllers; for example, to guarantee QUAVs stability and robustness during flight.

QUAVs require high-level designing and implementation of efficient multidimensional control structures that improve their response. Controllers rely on parameters that need to be tuned in order to obtain an adequate performance. Parameterization is based on iterative experiments of different trial-and-error methods. This process is time-consuming, and it does not guarantee getting optimal parameters.

The most popular controller types are proportional-derivative (PD) and proportional-integral-derivative (PID) because of their simplicity of structure and implementation. Unfortunately, there are very few systematic methods to find optimal parameters [4–6]. Nowadays, PID controls add techniques such as neural networks and evolutionary algorithms, resulting in high accuracy and dynamic performance for different engineering control problems. Systems are becoming adaptive to external disturbances. For example, in [7], an adaptive PID tuning method is reported, which is pre-trained by a genetic algorithm to tune the weights of a neural network and auto-tunes online the proportional, integral, and derivative units of a PID.

The design and implementation of control laws in QUAVs consider multiple variants that are tuned for each control in a constrained-based optimization manner, similar to the path planning problem [8,9]. Those limitations motivate solutions of stochastic nature. There are different stochastic algorithms, the most popular are metaheuristics, both evolutionary and population-based, with high effectiveness and easy implementation. Evolutionary metaheuristic algorithms are based on the laws of biological evolution, which do not share information with another generation. However, the best-qualified individuals composing the new group and better quality individuals will remain up to the end. Some of the most popular are genetic algorithm (GA) [10–12] and differential evolution (DE) [13] with many variants, refinements, and hybridization.

Population-based metaheuristics are motivated by collaboration capabilities, social interaction and sharing knowledge, present in many groups in nature. This type of metaheuristic is one of the most popular techniques to solve multiobjective optimization problems, thanks to their ability to compute multiple solutions in a single repetition [14].

The problem of tuning a PD or PID for a QUAV dynamic system results in a complex underactuated system, highly coupled in its variables, and many parameters to adjust, nonlinear, where it is not easy to apply conventional control techniques. Using these techniques, the dynamic model must meet convexity, differentiability, continuity, and linearity properties, leading to an excessive simplification of the model to obtain a global optimum, but failing to represent all the relevant features of the real system [15]. By using metaheuristic algorithms to solve this type of problem, it is impossible to show that the global optimum is reached; for this reason, a series of repetitions of the runs are carried out to verify the robustness and accuracy of the solutions obtained. Nevertheless, it generates satisfactory solutions in a reasonable time [16].

Previous studies have addressed the problem of tuning controllers applied to UAVs. In [17], a comparison between proportional (P), PD and PID controllers relating to time to tune and stability is done, applying a particle swarm optimization algorithm (PSO) for choosing the parameters to control attitude and altitude of QUAV (4 of 6 possible dynamics). Similarly, the authors in [18] have proposed the artificial bee colony (ABC) algorithm, PSO and GA to optimize PID parameters and add a comparison with Ziegler-Nichols method applied to QUAV attitude dynamics. In the same way, the optimization of a PID controller using GA, Crow Search Algorithm (CSA) and PSO only for orientation and altitude was performed in [19].

In this paper, we report the development of a dynamic model for a QUAV using the Newton–Euler formulation taking into account all the six dynamics for a QUAV. We compare eight optimization algorithms for PD and four for the PID controller. All algorithms were applied to the dynamic model. Initially, the parameters were obtained using different population-based metaheuristic algorithms, including PSO, Gray Wolf Optimization (GWO), Hunger Games Search (HGS), extended and improved SHADE (LSHADE, LSPACMA), Marine Predators Algorithm (MPA), Slime Mould Algorithm (SMA), and Whale Optimization Algorithm (WOA) [20–28].

Usually, a set of benchmark functions is used to demonstrate the performance of optimization algorithms by evaluating the accuracy, precision, and robustness. However, not all algorithms have the same performance capabilities for different engineering problems [29]. Thus, the main contribution of this work is to identify a set of metaheuristic algorithms that perform well in optimizing compensated PD and PID controls for trajectory tracking

of a QUAV considering the mathematical model is nonlinear with aerodynamic moments and forces on the Newton–Euler formulation. Consequently, obtaining an optimized and compensated structure that overcomes the limitations of nonlinearity and instability of a full dynamics (a total of six motions and directions) of a quadcopter system with PD or PID controls.

The mathematical model derived to simulate flight dynamics of a quadcopter and detail control strategy is given in Section 2. Section 3 details the optimization of compensated PD and PID. The evaluation methods and experimental results are presented and discussed in Section 4. Section 5 presents the conclusions and further outlooks in this paper.

2. Quadrotor UAV: Modeling and Control

This section presents the dynamical model of QUAV using the Newton–Euler formulation and detail control strategy.

2.1. System Description and Modeling of a QUAV

As shown in Figure 1, to describe the motion of QUAV in 3D space, this requires the definition of two frames: the inertial World frame, W , and the Body frame, B . The first is a fixed coordinate system on Earth with x_W – y_W – z_W and the second is fixed on the structure of the vehicle, denoted by x – y – z .

Figure 1. The QUAV and the reference frames.

The QUAV is a complex structure, and it involves different physical effects from aerodynamics and mechanic forces. The model developed assumes the following:

- The structure of the system is rigid and symmetrical.
- The gravity center and the B frame origin are coincident.
- Propellers are rigid.
- The attitude dynamics are limited to small angles and consequently slight variations of linear and angular velocities.

The absolute position of QUAV is defined as the coordinates in the B frame in regard to coordinates in the W frame, represented by position vector $\mathbf{r} = [x, y, z]^T$. The angular position, defined by the Euler angles (ϕ, θ, ψ) , represent the orientation of the QUAV with respect to the W frame, these angles define the rotations of *yaw*, *roll* and *pitch*, they indicate how the B frame (which initially is aligned with the W frame) must be rotated to align with W frame again.

The rotation matrix from B to W , is given by ${}^W R_B = {}^c R_B {}^a R_c {}^W R_a$, where ${}^c R_B$, ${}^a R_c$, ${}^W R_a$, represents rotations about X , Y and Z axis, respectively, in that order, as shown in Figure 2.

$${}^c R_B = \begin{pmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{aligned}
{}^a R_c &= \begin{pmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{pmatrix} \\
{}^W R_a &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{pmatrix} \\
{}^W R_B &= \begin{pmatrix} \cos(\theta)\cos(\psi) & \cos(\psi)\sin(\theta)\sin(\phi) - \cos(\phi)\sin(\psi) & \cos(\psi)\cos(\phi)\sin(\theta) + \sin(\psi)\sin(\phi) \\ \cos(\theta)\sin(\psi) & \cos(\psi)\cos(\phi) + \sin(\theta)\sin(\psi)\sin(\phi) & \cos(\phi)\sin(\theta)\sin(\psi) - \cos(\psi)\sin(\phi) \\ -\sin(\theta) & \cos(\theta)\sin(\phi) & \cos(\theta)\cos(\phi) \end{pmatrix}
\end{aligned}$$

Before presenting the movement equations, we establish the relation between velocities in W , $(\dot{\phi}, \dot{\theta}, \dot{\psi})$, and velocities in B , (p, q, r) , from transformation vectors:

$${}^W R_a {}^a R_c {}^c R_B = \begin{pmatrix} -s(\theta) \\ c(\theta)s(\phi) \\ c(\theta)c(\phi) \end{pmatrix},$$

$${}^W R_a {}^a R_c = \begin{pmatrix} 0 \\ c(\phi) \\ -s(\phi) \end{pmatrix},$$

$${}^W R_a = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix},$$

the rotational transform matrix is R_r

$$R_r = \begin{pmatrix} 1 & 0 & -s(\theta) \\ 0 & c(\phi) & c(\theta)s(\phi) \\ 0 & -s(\phi) & c(\theta)c(\phi) \end{pmatrix},$$

then, we can define the angular velocity in the body, ω_B , as the relation

$$\omega_B = \begin{pmatrix} p \\ q \\ r \end{pmatrix} = R_r \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix},$$

by using a small angle approximation, we get

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} p + r \cos(\phi) \tan(\theta) + q \sin(\phi) \tan(\theta) \\ q \cos(\phi) - r \sin(\phi) \\ r \cos(\phi) \sec(\theta) + q \sin(\phi) \sec(\theta) \end{pmatrix} \approx \begin{pmatrix} p \\ q \\ r \end{pmatrix}.$$

Applying, Newton–Euler formulation, equations of motion are given by (1) and (2).

$$m\dot{\mathbf{v}} = \mathbf{F}_g + \mathbf{F}_d + \mathbf{F} \quad (1)$$

$$\mathbf{I}_B \dot{\omega}_B + (\omega_B \times \mathbf{I}_B \omega_B) + \mathbf{M}_a = \mathbf{M}_B \quad (2)$$

In Equation (1), the acceleration of mass ($m\dot{\mathbf{v}} = m(\ddot{x} \ \ddot{y} \ \ddot{z})^T$), depends on gravity force ($\mathbf{F}_g = (0, 0, -mg)^T$), drag forces ($\mathbf{F}_d = \text{diag}(k_1, k_2, k_3)\mathbf{v}^T$) and the total thrust of the four rotors ($\mathbf{F} = {}^W R_B[0, 0, u_1]^T$). In Equation (2), the angular acceleration of the inertia ($\mathbf{I}_B \dot{\omega}_B$), the centripetal forces ($(\omega_B \times \mathbf{I}_B \omega_B)$) and the aerodynamic forces ($\mathbf{M}_a = \text{diag}(k_4, k_5, k_6)\omega_B^2$), are equal to the external torque ($\mathbf{M}_B = (u_2, u_3, u_4)^T$). Substituting forces expressions and each moment into (1) y (2) we have the following translational and rotational dynamics of the QUAUV:

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{1}{m}(\cos(\psi)\sin(\theta)\cos(\phi) + \sin(\phi)\sin(\psi))u_1 - \frac{k_1}{m}\dot{x} \\ \frac{1}{m}(\sin(\psi)\sin(\theta)\cos(\phi) - \sin(\phi)\cos(\psi))u_1 - \frac{k_2}{m}\dot{y} \\ \frac{1}{m}(\cos(\phi)\cos(\theta))u_1 - g - \frac{k_3}{m}\dot{z} \\ \frac{(I_y - I_z)}{I_x}\dot{\theta}\dot{\psi} - \frac{k_4}{I_x}\dot{\phi}^2 + \frac{1}{I_x}u_2 \\ \frac{(I_z - I_x)}{I_y}\dot{\phi}\dot{\psi} - \frac{k_5}{I_y}\dot{\theta}^2 + \frac{1}{I_y}u_3 \\ \frac{(I_x - I_y)}{I_z}\dot{\phi}\dot{\theta} - \frac{k_6}{I_z}\dot{\psi}^2 + \frac{1}{I_z}u_4 \end{bmatrix}. \quad (3)$$

Figure 2. Order of rotations from W to B.

2.2. Strategy Control of Trajectory Tracking

In order for the QUAUV to follow a predetermined path in three-dimensional space, the control of trajectory tracking is proposed as Figure 3, where we introduce the desired positions x_d, y_d, z_d to the Translational and Altitude Controller, which computes the control signal u_1 and determines the desired angles ϕ_d, θ_d , which in turn are the references to the Attitude Controller. The Attitude Controller calculates the control signals u_2, u_3 and u_4 for orientation dynamics.

Figure 3. Proposed control structure to trajectory tracking.

Due to the nonlinear nature of QUAUV system, a compensated PID and PD controller for trajectory tracking is proposed [30].

2.2.1. PD Controller

The dynamics shown in (3) can be represented as a second-order system, in the form

$$\ddot{\chi} = f(\chi) + g(\chi)u, \quad (4)$$

where $f(\chi)$, in this work, it is a nonlinear function due to the drag forces of the QUAV, $\chi = [\chi, \dot{\chi}]$, $g(\chi) \neq 0$ is a known function u , is the control signal. Control law is designed as

$$u = \frac{1}{g(\chi)} \left(-f(\chi) + K^T E + \ddot{\chi}_d \right), \quad (5)$$

where, $\ddot{\chi}_d$ is the second derivative of the desired trajectory, $E = [e \ \dot{e}]^T$, with $e = \chi_d - \chi$ substituting (5) in (4), the closed-loop dynamics of the system are

$$\ddot{e} + k_p e + k_d \dot{e} = 0,$$

- Translational and Altitude Control

The altitude dynamics shown in (3), can be represented as (4),

$$\ddot{z} = \frac{1}{m} (\cos \phi \cos \theta) u_1 - g - \frac{k_3}{m} \dot{z}, \quad (6)$$

Thus, if the control signal is designed as in (5) we get

$$u_1 = \frac{m}{\cos \theta \cos \phi} \left(\frac{k_3}{m} \dot{z} + K_z^T E_z + \ddot{z}_d \right) + g, \quad (7)$$

Equation (7) is constrained $(-\pi/2 < \phi < \pi/2)$ and $(-\pi/2 < \theta < \pi/2)$.

For translation dynamic in the axis X , a virtual control signal is selected as $\mu_x = \cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi$, then becomes as

$$m\ddot{x} = f(x) + \mu_x u_1, \quad (8)$$

and the control signal is chosen as in (5)

$$\mu_x = \frac{m}{u_1} \left(\frac{k_1}{m} \dot{x} + K_x^T E_x + \ddot{x}_d \right). \quad (9)$$

Similarly, for the dynamic on the Y axis, $\mu_y = \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi$, we get

$$m\ddot{y} = \frac{k_2}{m} \dot{y} + \mu_y u_1, \quad (10)$$

with the control signal as form as (5), therefore

$$\mu_y = \frac{m}{u_1} \left(\frac{k_2}{m} \dot{y} + K_y^T E_y + \ddot{y}_d \right). \quad (11)$$

Equations (9) and (11) are true for $u_1 > 0$, which is physically correct because u_1 represents the lift force generated by the rotors of the vehicle.

Virtual control signals μ_x y μ_y allow to calculate desired angles for attitude controller $roll$, ϕ_d y $pitch$, θ_d :

$$\begin{aligned} \begin{bmatrix} \cos(\phi) \sin(\theta) \\ \sin(\phi) \end{bmatrix} &= \begin{bmatrix} \cos(\psi) & \sin(\psi) \\ \sin(\psi) & -\cos(\psi) \end{bmatrix} \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix} \\ \phi_d &= \arcsin(\sin(\psi)\mu_x - \cos(\psi)\mu_y) \\ \theta_d &= \arcsin\left(\frac{\cos(\psi)\mu_x + \sin(\psi)\mu_y}{\cos(\phi)}\right). \end{aligned} \quad (12)$$

- Attitude Controller

For the orientation dynamics, they can be represented as (4) and the control signal can be designed as in (5).

For the movement of *roll*, we have the control signal u_2 as:

$$u_2 = I_x \left(\frac{k_4}{I_x} \dot{\phi}^2 - \frac{(I_y - I_z)}{I_x} \dot{\theta} \dot{\psi} + \mathbf{K}_\phi^T E_\phi + \ddot{\phi}_d \right), \quad (13)$$

For the movement of *pitch*, we have the control signal u_3 as:

$$u_3 = I_y \left(\frac{k_5}{I_y} \dot{\theta}^2 - \frac{(I_z - I_x)}{I_y} \dot{\phi} \dot{\psi} + \mathbf{K}_\theta^T E_\theta + \ddot{\theta}_d \right), \quad (14)$$

Finally, for the movement of *yaw*, we have the control signal u_4 as:

$$u_4 = I_z \left(\frac{k_6}{I_z} \dot{\psi}^2 - \frac{(I_x - I_y)}{I_z} \dot{\phi} \dot{\theta} + \mathbf{K}_\psi^T E_\psi + \ddot{\psi}_d \right). \quad (15)$$

2.2.2. PID Controller

The PID controller is designed in the same way as the PD controller. The difference lies in the addition of a variable in the error E and in the gain vector K . They both have the same structure. We recall control law in (5)

$$u = \frac{1}{g(\chi)} \left(-f(\chi) + \mathbf{K}^T E + \ddot{\chi}_d \right),$$

where, $\ddot{\chi}_d$ is the second derivative of the desired trajectory. Now, $E = [e \int e \dot{e}]^T$, with $e = \chi_d - \chi$ and the closed-loop dynamics of the system are

$$\ddot{e} + k_p e + k_i \int e + k_d \dot{e} = 0.$$

3. Optimization of Compensated Controllers

Figure 4, shows the implemented scheme for the controller parameterization, where each of the metaheuristic algorithms is evaluated. Initially, the population of the algorithm is generated and evaluated. The algorithm delivers a set of gains, which are applied to the designed controller, which in turn calculates the control signals for the QUAV. Then, the dynamic performance of each of the X, Y and Z axes is measured to calculate the error between the desired dynamics and the real one. This cycle is repeated until the number of iterations initially determined is reached or until the error or cost function reaches a set value.

To obtain the parameters of each of the six UAV dynamics, through the evolutionary algorithms, the population to optimize the PD controller is:

$$P = \{k_{p_x} k_{d_x} k_{p_y} k_{d_y} k_{p_z} k_{d_z} k_{p_\phi} k_{d_\phi} k_{p_\theta} k_{d_\theta} k_{p_\psi} k_{d_\psi}\}, \quad (16)$$

and the population to optimize the PID controller is:

$$P = \{k_{p_x} k_{i_x} k_{d_x} k_{p_y} k_{i_y} k_{d_y} k_{p_z} k_{i_z} k_{d_z} k_{p_\phi} k_{i_\phi} k_{d_\phi} k_{p_\theta} k_{i_\theta} k_{d_\theta} k_{p_\psi} k_{i_\psi} k_{d_\psi}\}, \quad (17)$$

with the search space for each individual $S = [0, 10]$, chosen that way to restrict the values of the control signals due to the physical limitations of the actuators when performing platform experiments. The cost function calculated using the root mean square error:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^G \sum_{k=0}^N (\chi_d^i(k) - \chi^i(k))^2}, \quad (18)$$

where N is the size of χ and G , represents each of the dynamics $x, y, z, \phi, \theta, \psi$ of QUAV.

Figure 4. Flowchart: controller parameter tuning process.

4. Results and Discussion

We validated a dynamic model through simulations that verified the behavior of coupled dynamics and no linearity. Calculations were performed in a Mac OS Big Sur (Intel core i9, RAM 24 GB, and 1 TB hard disk), using MATLAB 2015a (MathWorks) for each algorithm (PSO, GWO, HGS, LSHADE, SPACMA, MPA, SMA, WOA). The solution space for each parameter to optimize (total 12, Equation (16)) was constrained to [0,10]. We performed simulations for 20, 30, and 40 individuals using 20 iterations during 30 independent runs for PD controllers. We ran simulations with 80 and 100 individuals for PID controllers using 40 iterations and 30 independent runs. The evaluation performance of the algorithms is reported through statistics that considers four variables to check the dispersion or variability of the solutions, which are the best, the mean, the standard deviation and the worst. In the following tables, the best results are presented in bold, and the purpose is to quickly identify the best performing controller as well as its gains.

4.1. PD Controller

Table 1 shows the RMSE statistics for the simulation with 20 individuals, 20 evolutions and 30 repetitions of the eight algorithms under test. The HGS algorithm is the one that obtains the best value $RMSE = 0.037263535506379$, the MPA algorithm is the one that obtains the best average, standard deviation and worst values. Figure 5a show the RMSE during the 20 iterations, the three best values are HGS, MPA and PSO. In the box plot, the lowest variability is obtained by GWO and MPA and the highest variability of the RMSE value is in PSO algorithm. The 12 parameters identified by each algorithm are shown in Table 2, where the HGS parameters are the best for obtaining a lower RMSE.

Table 3 contains the RMSE statistics for the simulation with 30 individuals, 20 iterations and 30 repetitions. The algorithm that obtains the best performance is again the HGS with a $RMSE = 0.03725093743996$; for this case, the GWO algorithm gets the best average, standard deviation and worst values. Figure 5b show the RMSE during the 20 iterations; the three best values are HGS, PSO and MPA. In the box plot, the lowest variability is

obtained by GWO, HGS, MPA y SMA and the highest variability of the RMSE value is obtained by PSO algorithm. The 12 parameters identified by each algorithm are shown in Table 4. The HGS parameters are the best for getting the lowest RMSE.

Table 5 contains the RMSE statistics for the simulation of 40 individuals, 20 evolutions and 30 runs with the corresponding algorithms. The algorithm that obtains the best performance is the HGS with a $RMSE = 0.037247252379126$. The MSA algorithm gets the best average, standard deviation, and worst values for this case. The 12 parameters identified by each algorithm are shown in Table 6. In Figure 5c, the box plot shows the lowest variability is obtained by GWO, HGS, MPA and SMA. The highest variability of the RMSE value is obtained by PSO algorithm.

We evaluated the performance of optimized parameters obtained (Table 2) for all six dynamics, $(x, y, z, \phi, \theta, \psi)$ and algorithms (Figures 6–11) at 20 iterations with 20 individuals. For translational dynamics (x and y), the desired trajectory is trapezoidal and for altitude (z) and yaw (ψ) dynamics is a ramp. The sharp effect observed in the Z axis of Figure 8 is due to the compensation between the force of gravity and the control signal. Negative values are associated with the instant in which the value of gravity is more significant than the control signal. Please note that this effect is only visible in the simulation; in real life, this effect is observed as a time delay until the force of gravity is countered by the control signal. The best trajectory is determined by the minimal value of cost function achieved by HGS as shown in Table 1.

The highest cost function value will define an acceptable performance for all dynamics. Here, the algorithm LSPACMA reached the highest value (0.040248443990318). As we can see, HGS algorithms performed best with a minimum square error (RMSE), which is the lowest obtained for all algorithms. LSPACMA has the largest error in most of the dynamics performance.

Figures 9 and 10 show the angles (ϕ and θ) of orientation of the QUAV. Dotted lines show the desired angles calculated as described in Section 2.2.1. Simulations show that ϕ angles are coupled with the axis Y and rotational movements of θ are coupled with axis X . When a change in direction is necessary, the controls are robust enough to generate changes in the orientation in response to the desired angles.

Figure 12 shows the control signals calculated by (7), (13)–(15). It is noted that the controllers reach their steady states during the flight. Signals u_2 and u_3 respond to changes in direction on the desired path and u_1 is affected and responds to small variations in altitude. The dynamic of ψ converges in minimum time and consequently, the control signal u_4 is kept at a small value.

In Figure 13, we can see that the trajectory tracking in X – Y – Z space, evaluating the optimized parameters, is successfully developed. With the obtained parameters, we consider a different circuit. The desired trajectory is sinusoidal for translational dynamics (x and y), and altitude (z) is a step. We found that the QUAV can follow it, as shown in Figure 14.

Table 1. Experimental result of RMSE with 20 individuals, 20 evolutions and 30 repetitions.

Statistics	PSO	GWO	HGS	LSHADE	LSPACMA	MPA	SMA	WOA
Best	0.037396168112890	0.037362955856019	0.037263535506379	0.038978058766757	0.039741105315137	0.037284239891813	0.037835061121072	0.037370940741738
Mean	0.054376357228900	0.043954323277539	0.045379957656500	0.052582989549809	0.050922002950839	0.038978646154916	0.051468628827586	0.068998262466992
SD	0.018193244089308	0.018827354037126	0.012805236930367	0.014038286466383	0.008042296173043	0.001885601376063	0.017924632527572	0.103765679125065
Worst	0.074792336225780	0.140069824231860	0.074527071012758	0.091325532418241	0.079482210630274	0.044105573338656	0.107284349261279	0.615170818395657

Table 2. The parameter estimation of metaheuristic algorithms with 20 individuals, 20 evolutions and 30 repetitions

Parameters	PSO	GWO	HGS	LSHADE	LSPACMA	MPA	SMA	WOA
k_{p_x}	5.408936203613887	5.213781792960648	5.412177273755431	5.715231175228559	6.620122525285044	5.481978926698987	5.711436859433793	5.437501487917403
k_{d_x}	10.000000000000000	10.000000000000000	10.000000000000000	9.325716854587114	9.993641853644958	9.999863024805832	10.000000000000000	10.000000000000000
k_{p_y}	5.096583444010690	4.891463309029316	5.368888612384497	4.877734569229675	3.772435488131499	4.896793864900809	4.548312153197413	6.315113281984011
k_{d_y}	10.000000000000000	10.000000000000000	10.000000000000000	9.800086546316058	9.678338787974745	10.000000000000000	10.000000000000000	10.000000000000000
k_{p_z}	10.000000000000000	10.000000000000000	9.950128906195692	9.590718062002178	6.771555264587359	9.958555777339498	9.771978015658293	10.000000000000000
k_{d_z}	8.285393139094424	8.260159603944022	10.000000000000000	9.722104767452858	9.101622173737571	10.000000000000000	5.351259401838421	10.000000000000000
k_{p_ϕ}	7.058540895556090	4.822381113908009	4.709902665404981	1.427050424025854	9.962956330944161	6.546159756777394	2.246128817422761	10.000000000000000
k_{d_ϕ}	4.943026175684162	8.194215866646751	5.068174591175525	2.123783954799148	3.186248449818443	9.905810957096076	4.860624932515171	10.000000000000000
k_{p_θ}	4.844306519002986	4.784138401376938	4.821175176721279	4.895525445351029	5.247032886854611	4.916081555171791	5.047194369837532	4.711197637806473
k_{d_θ}	0	0	0.000035339909776	0.000244462522990	0.060372557625467	0.000127362385910	0.00009105416033	0
k_{p_ψ}	5.355372158301154	4.069504903320026	0.005334006148015	3.305332611822569	5.476771414363572	7.773619983493155	0.949746659092169	10.000000000000000
k_{d_ψ}	3.808860161714160	0.175196233025389	0.043112681853079	0.317132270794582	7.891062763173645	9.976206665270322	0.361836747670447	8.120054392303674

Table 3. Experimental result of MSE with 30 individuals, 20 evolutions and 30 repetitions.

Statistics	PSO	GWO	HGS	LSHADE	LSPACMA	MPA	SMA	WOA
Best	0.037298689699719	0.037386364140826	0.037250937439961	0.038703382913285	0.039391601057574	0.037305699738380	0.037610074434848	0.037526810582112
Median	0.049292829641084	0.038231926452584	0.181468520773043	0.048098319555092	0.048447470240744	0.038439358577506	0.040600753887036	0.106439604511052
SD	0.016922702608194	0.000917745219989	0.480087752312157	0.008671308416183	0.007749711491518	0.001368117384585	0.006458484603516	0.164512311240661
Worst	0.074597379399439	0.040928476103477	1.972825126787748	0.077406765826570	0.078783202115148	0.043410168070117	0.066852769977100	0.608263592411619

Table 4. The parameter estimation of metaheuristic algorithms with 30 individuals, 20 evolutions and 30 repetitions.

Parameters	PSO	GWO	HGS	LSHADE	LSPACMA	MPA	SMA	WOA
k_{p_x}	5.321480404491486	5.316121848678594	5.506178152902620	5.573040492118754	6.456895021660555	5.597288673172770	5.357830158921898	5.985862339317787
k_{d_x}	10.000000000000000	10.000000000000000	10.000000000000000	9.781398388783654	9.838517000280444	9.999480002823272	10.000000000000000	10.000000000000000
k_{p_y}	5.360362063749348	5.022574636552078	5.575896169072749	7.045066726522021	6.373778110323427	5.777679946622201	5.492717168284171	5.831980752133919
k_{d_y}	10.000000000000000	10.000000000000000	10.000000000000000	9.732603652525405	9.441276780545653	10.000000000000000	10.000000000000000	10.000000000000000
k_{p_z}	10.000000000000000	9.127393279440687	10.000000000000000	8.640471589965724	8.362856784741542	10.000000000000000	9.982166969934834	10.000000000000000
k_{d_z}	9.467313952657385	8.846939435077033	10.000000000000000	8.419482408166351	9.849967164971744	9.958303650280419	6.636230464386927	10.000000000000000
k_{p_ϕ}	6.814063438272202	3.228097695082624	4.541639604890506	6.863481745971876	8.736217915815690	9.847393024662287	10.000000000000000	5.796906949590673
k_{d_ϕ}	5.874995512822094	8.203547303875149	7.158872343937848	5.435527296456358	4.736415620880451	9.994682117338073	5.067661699504412	10.000000000000000
k_{p_θ}	4.819432444500308	4.774213565304394	4.861996886493411	5.066511011257896	4.508383434244966	4.900504250374575	4.697810473694974	4.624341847012862
k_{d_θ}	0	0	0	0.021053555495621	0.012780497660689	0	0	0
k_{p_ψ}	6.781239359799283	0.961791842657223	0.814480987696878	6.972615636034867	3.964876907491253	9.875696545163565	6.784589697287003	4.717551680873766
k_{d_ψ}	6.535849274794208	2.450515048981552	6.353764374969330	1.691008581444079	8.632031591890170	9.950887888897011	0	4.594521890201812

Table 5. Experimental result of RMSE with 40 individuals, 20 evolutions and 30 repetitions.

Statistics	PSO	GWO	HGS	LSHADE	LSPACMA	MPA	SMA	WOA
Best	0.037255373533488	0.037266015064895	0.037247252379126	0.039394303548209	0.038830921234619	0.037256592611999	0.037632310844408	0.037353819712401
Mean	0.045525283051302	0.041601797209679	0.066221944130588	0.043549315402296	0.045455785193484	0.038150710592382	0.039375241940074	0.043928558596090
SD	0.012064390608393	0.010639844056084	0.154265972701073	0.003114729193649	0.005768092569169	0.001152641167111	0.002176578772137	0.005084363577577
Worst	0.074392122335153	0.084154455562805	0.882938952672554	0.051704889477973	0.062767121112513	0.043218862736804	0.045909805573245	0.059239414200638

Table 6. The parameter estimation of metaheuristic algorithms with 40 individuals, 20 evolutions and 30 repetitions.

Parameters	PSO	GWO	HGS	LSHADE	LSPACMA	MPA	SMA	WOA
k_{p_x}	5.387775921956180	5.355284009147565	5.381403859043313	5.513337891603976	5.234732646639856	5.425915523532709	4.965902929744495	5.821726954392787
k_{d_x}	10.000000000000000	10.000000000000000	10.000000000000000	9.933746539383339	9.733042357851929	10.000000000000000	10.000000000000000	10.000000000000000
k_{p_y}	5.503169448530415	5.157652533295196	5.385842705691601	7.630413711776860	5.066585315120156	5.431085259751310	6.231850241856680	5.339150052042956
k_{d_y}	10.000000000000000	10.000000000000000	10.000000000000000	9.112575074225544	9.601563597114865	9.999805203324547	10.000000000000000	10.000000000000000
k_{p_z}	10.000000000000000	10.000000000000000	10.000000000000000	8.452040812542801	8.276025623946069	10.000000000000000	9.757836996055245	10.000000000000000
k_{d_z}	10.000000000000000	10.000000000000000	9.999937779326700	7.857006737354142	8.533283889078021	9.989924454194904	9.979985496334406	10.000000000000000
k_{p_ϕ}	5.509435611867958	6.302627444914116	6.386994221918801	7.928849397796594	7.473150998878030	8.009573766497542	4.558718104204019	10.000000000000000
k_{d_ϕ}	9.432035226553529	8.636335189288980	9.788907844769838	8.628408426770227	9.597213064062426	9.678365436395564	1.593956521679671	10.000000000000000
k_{p_θ}	4.794923996582154	4.910674065557497	4.826014474662918	5.075458800622620	4.630441901688464	4.823227320881963	4.753962189265832	4.763044802438290
k_{d_θ}	0	0	0	0.008017685879173	0.027765559630236	0.000003518624809	0	0
k_{p_ψ}	7.511633082929468	3.018192575071655	1.766328272277126	1.903434392477181	8.539208296636906	9.998471291768176	9.814218990526523	9.362558661978090
k_{p_ψ}	7.963122860297110	3.360333706177101	0.191715360093252	4.441115912390526	6.058651260609002	8.920397598071631	3.400723395395384	8.585227416173138

Figure 5. Convergence curves of RMSE and boxplot distribution of the different algorithms, iterations and individuals. (a) 20 individuals, 20 iterations; (b) 30 individuals, 20 iterations; (c) 40 individuals, 20 iterations.

Figure 6. Comparison of tracking performance in X-axis of the different algorithms evaluating parameters obtained with 20 individuals.

Figure 7. Comparison of tracking performance in Y-axis of the different algorithms evaluating parameters obtained with 20 individuals.

Figure 8. Comparison of tracking performance in Z-axis of the different algorithms evaluating parameters obtained with 20 individuals.

Figure 9. Comparison of tracking performance in ϕ -angle of the different algorithms evaluating parameters obtained with 20 individuals.

Figure 10. Comparison of tracking performance in θ -angle of the different algorithms evaluating the parameters obtained with 20 individuals.

Figure 11. Comparison of tracking performance in ψ -angle of the different algorithms evaluating the parameters obtained with 20 individuals.

Figure 12. Signals (u_1, u_2, u_3, u_4), calculated by PD control optimized with 100 individuals.

Figure 13. Tracking performance in three-dimensional space with optimized parameters using HGS algorithm.

Figure 14. Tracking performance in three-dimensional space changing desired trajectory.

4.2. PID Controller

From the previous section, the four algorithms with the best fitness during tuning of the PD controls were GWO, HGS, MPA and SMA; these are taken for the PID control. Eighteen parameters were identified (Equation (17)), so it was necessary to increase the number of individuals. Here, two simulations are reported, one with 80 and one with 100 individuals. The number of evolutions and repetitions is 40 and 30, respectively.

Table 7 shows the RMSE statistics for the simulation with 80 individuals, 40 evolutions and 30 repetitions of the four algorithms under test. The HGS algorithm obtained the best value ($RMSE = 0.032594329329481$). We also show the best values for all algorithms for the mean, standard deviation, and worst. Figure 15a shows the RMSE during the 40 iterations. The three best values are HGS, MPA and GWO. In the box plot, the lowest variability is obtained by GWO, HGS, and SMA. The highest variability of the RMSE value is in MPA algorithm. The 18 parameters identified by each algorithm are shown in Table 8, where the HGS parameters are the best for obtaining a lower RMSE

Table 9 contains the RMSE statistics for the simulation with 100 individuals, 40 iterations and 30 repetitions. The algorithm that obtains the best performance is the HGS with a $RMSE = 0.032594309723623$. The GWO algorithm obtains the best average and standard deviation for this case. The MPA obtains the worst values. Figure 15b show the RMSE during the 40 iterations. The three best values are HGS, GWO and MPA. GWO, HGS, and SMA obtain the lowest variability in the box plot. This graph shows a smaller dispersion of the optimization solutions compared to the simulation with 80 individuals. The 18 parameters identified by each algorithm are shown in Table 10. The HGS parameters are the best for obtaining the lowest RMSE. The 18 parameters identified by each algorithm are shown in Table 8, where the HGS parameters are the best for obtaining a lower RMSE.

We evaluated the performance of optimized parameters obtained (Table 10) for all six dynamics, $(x, y, z, \phi, \theta, \psi)$ and algorithms (Figures 16–21) at 40 iterations with 100 individuals. The desired trajectories are the same as for the PD controller experiment. The best trajectory is determined by the minimal value of cost function achieved by HGS as shown in Table 9. The similarity of the performance of the six dynamics with the PD controller implementation can be noted, even when the number of parameters to optimize is larger and the number of individuals is bigger.

Figure 22 shows that the signals calculated by the PID controllers stabilize the dynamics. The control signals are very similar to Figure 12 since the desired path is the same. In the same way, as in the PD control, signals u_2 and u_3 respond to changes in direction on the desired path and u_1 is affected and responds to small variations in altitude. The dynamic of ψ converges in minimum time and consequently, the control signal u_4 is kept at a small value.

Finally, Figure 23 shows the successful trajectory tracking in X–Y–Z space, evaluating the optimized parameters of PID controller using HGS algorithm with 100 individuals. With the same optimized parameters, we consider a different path. The desired trajectory is a sinewave for translational dynamics (x and y), and altitude (z) is a step. We found that the QUAV can perform it effectively, as shown in Figure 24.

Table 7. Experimental result of RMSE with 80 individuals, 40 evolutions and 30 repetitions.

Statistics	GWO	HGS	MPA	SMA
Best	0.032952255627613	0.032594329329481	0.032886465790112	0.035126701196436
Mean	0.039467436415521	0.038750257627602	0.060565527073854	0.042743594234625
SD	0.024810306002939	0.017270207475356	0.070036980113846	0.036517035201977
Worst	0.188774298939131	0.108450087893772	0.399767783188887	0.264990988241978

Table 8. The parameter estimation of metaheuristic algorithms with 80 individuals, 40 evolutions and 30 repetitions.

Parameters	GWO	HGS	MPA	SMA
k_{p_x}	9.98652117275694	10	10	9.99977970933963
k_{i_x}	9.43065757230055	9.65685137184573	9.96059244300613	9.99925593925136
k_{d_x}	5.60150169731773	5.47885679524568	5.5092048396275	6.2452264065116
k_{p_y}	10	9.99999921053496	10	9.99246289808091
k_{i_y}	9.63952866608376	10	10	9.43597394420227
k_{d_y}	4.95091215636854	5.11571890362846	5.27988640351169	7.32911087094447
k_{p_z}	9.87615694650701	9.99999863024639	10	4.46530734475299
k_{i_z}	2.15795493555051	0	9.97237668436031	0
k_{d_z}	9.98736121621257	10	9.9995314190582	10
k_{p_ϕ}	3.01042546564222	2.22250978351406	5.85696172264884	1.95386097208097
k_{i_ϕ}	9.94312800230129	10	10	8.63779129530408
k_{d_ϕ}	7.12597992811241	7.15473933164715	8.20505265096145	9.67217493667522
k_{p_θ}	4.72796195352608	4.71467947432073	4.7240318855983	4.55973192469571
k_{i_θ}	0.0544330911469697	0	0	0.0492159899582777
k_{d_θ}	0.0126001912603102	0	0	0.0177826862585763
k_{p_ψ}	7.76164310613195	0.462448362031975	0.0542303149109458	5.78562381917693
k_{i_ψ}	3.72469145930572	3.82226368134878	9.98533786473978	0.488908932241553
k_{d_ψ}	4.53181675746977	5.98213376475638	9.97253061093644	10

Table 9. Experimental result of RMSE with 100 individuals, 40 evolutions and 30 repetitions.

Statistics	GWO	HGS	MPA	SMA
Best	0.032996145705117	0.032594309723623	0.03282576420999	0.040673064569420
Mean	0.039979172065477	0.042389401130873	0.043522145600419	0.043522145600419
SD	0.018550824347545	0.033123701556092	0.017238485044736	0.026360509636230
Worst	0.108674487023002	0.180699757796690	0.091285974784260	0.205159863683217

Table 10. The parameter estimation of metaheuristic algorithms with 100 individuals, 40 evolutions and 30 repetitions.

Parameters	GWO	HGS	MPA	SMA
k_{p_x}	10	10	10	10
k_{i_x}	9.67154727244253	9.68792081138642	9.92750116646622	0.0503735259587369
k_{d_x}	5.55507505212891	5.48183670919142	5.4719400159824	2.97829672909644
k_{p_y}	10	10	10	9.26519695366179
k_{i_y}	9.94398012231298	10	10	0.027492447132257
k_{d_y}	4.90915262148882	5.11509775200818	5.07430677163768	7.32731572860309
k_{p_z}	10	10	10	10
k_{i_z}	6.24600523295865	0	9.9847000333801	0.501988975857219
k_{d_z}	10	10	10	6.96510555267074
k_{p_ϕ}	7.20349062244415	2.24353462117906	2.6508561094601	10
k_{i_ϕ}	6.2669806605969	10	6.0368191032762	0
k_{d_ϕ}	5.3959150491106	7.15275868300352	4.55665073094864	8.90239259865836
k_{p_θ}	4.73709682183962	4.71756871088933	4.71221992462718	4.52648291628
k_{i_θ}	0.0326031028012331	0	1.51704532400754e-07	0
k_{d_θ}	0	0	1.92350263205458e-05	0
k_{p_ψ}	3.88522491995354	0.00013406587418094	9.85486072057678	10
k_{i_ψ}	0.191594435931627	0.133663724883617	9.99867297798012	0
k_{d_ψ}	3.19550238953637	10	9.99637332392654	5.68443608990096

Figure 15. Convergence curves of RMSE and boxplot distribution of the different algorithms, 40 iterations and different individuals. (a) 80 individuals, 40 iterations; (b) 100 individuals, 40 iterations.

Figure 16. Comparison of tracking performance in X-axis of the different algorithms evaluating parameters obtained with 100 individuals.

Figure 17. Comparison of tracking performance in Y-axis of the different algorithms evaluating parameters obtained with 100 individuals.

Figure 18. Comparison of tracking performance in Z-axis of the different algorithms evaluating parameters obtained with 100 individuals.

Figure 19. Comparison of tracking performance in ϕ -angle of the different algorithms evaluating parameters obtained with 100 individuals.

Figure 20. Comparison of tracking performance in θ -angle of the different algorithms evaluating parameters obtained with 100 individuals.

Figure 21. Comparison of tracking performance in ψ -angle of the different algorithms evaluating parameters obtained with 100 individuals.

Figure 22. Signal response (u_1, u_2, u_3, u_4) from PID controller optimized with 100 individuals.

Figure 23. Tracking performance in three-dimensional space with optimized parameters of PID controller using HGS algorithm with 100 individuals.

Figure 24. Tracking performance in three-dimensional space changing desired trajectory. The optimized parameters of PID controller obtained and HGS algorithm with 100 individuals.

5. Conclusions

In this paper, we have addressed a long-standing problem related to the tuning of compensated PD and PID controllers for path tracking of the QUAV vehicle modeled with six different dynamics. We compared eight metaheuristic algorithms (PSO, GWO, HGS, LSHADE, LSPACMA, MPA, SMA and WOA), finding optimal parameters to select the most accurate algorithm. The HGS algorithm was the most suitable of the eight evaluated, as it obtained the lowest RMSE for the compensated PD and PID controls. The trajectory used was a square with an elevation change, with 40 individuals and 20 evolutions for the compensated PD control, $RMSE = 0.03724725237912$, and the number of optimized parameters is twelve. For the compensated PID control with 100 individuals and 40 evolutions, the $RMSE = 0.032594309723623$, the number of optimized parameters is eighteen. For the case of compensated PD control, the following best algorithms are GWO, MPA and SMA, with an RMSE close to HGS. This case corresponds to when 40 individuals are used, and for the case of compensated PID control, only the HGS, GWO, MPA and SMA were simulated. On the other hand, the algorithm LSPACMA was the most limited in optimization scope. We also found those optimized gains for the square path could take the QUAV to a different trajectory like the sinewave and step paths, resulting in the most suitable tuning for the compensated PD and PID controllers for QUAV trajectory tracking. This modeling approach is now ready and suited for implementation at an experimental level.

Author Contributions: Conceptualization, N.S.Z.-P., N.H.-R. and J.C.S.-T.-M.; methodology, N.S.Z.-P., N.H.-R. and J.C.S.-T.-M.; software, N.S.Z.-P. and N.H.-R.; validation, N.S.Z.-P., N.H.-R. and I.B.-V.; formal analysis, N.S.Z.-P., N.H.-R. and J.C.S.-T.-M.; investigation, N.S.Z.-P., N.H.-R. and J.C.S.-T.-M.; resources, J.M.-M. and J.C.S.-T.-M.; data curation, N.S.Z.-P.; writing—original draft preparation, N.S.Z.-P. and I.B.-V.; writing—review and editing, N.S.Z.-P. and I.B.-V.; visualization, N.S.Z.-P. and I.B.-V.; supervision, J.M.-M. and J.C.S.-T.-M.; project administration, J.M.-M. and J.C.S.-T.-M.; funding acquisition, J.M.-M. and J.C.S.-T.-M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Council for Science and Technology (CONACYT) with project number CB- 2017-2018-A1-S-43008. Nadia Samantha Zuñiga Peña was supported by CONACYT grant number 785376.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Desired trajectories and dynamic performance data are publicly available at <https://github.com/NadiaZunigaPena/Metaheuristics-enabled-path-tracking-Optimization-in-Unmanned-Aerial-Vehicles>, accessed on 26 October 2021.

Conflicts of Interest: The authors declare that they have no known competing financial interest or personal relationships that could have appeared to influence the work reported in this paper.

Nomenclature

Symbol	Description
x, y, z	Position coordinates respect to World frame
$\dot{x}, \dot{y}, \dot{z}$	Linear velocity respect to World frame
$\ddot{x}, \ddot{y}, \ddot{z}$	Linear acceleration respect to World frame
ϕ, θ, ψ	Attitude coordinates respect to World frame
$\dot{\phi}, \dot{\theta}, \dot{\psi}$	Angular velocity respect to World frame
$\ddot{\phi}, \ddot{\theta}, \ddot{\psi}$	Angular acceleration respect to World frame
${}^W R_B$	Rotation matrix for Body frame to World frame
p, q, r	Angular velocity respect in Body frame
$\dot{p}, \dot{q}, \dot{r}$	Angular acceleration respect in World frame
g	Gravity
m	Mass of QUAV

M_a	Moments resulting from aerodynamic frictions
M_b	Moments generated by motors respect to mass center of vehicle.
I_B	Matrix of inertia moments
F_B	Vector of thrust
F_d	Vector of drag forces
F_g	Vector of forces due to gravity
$k_{1,2,\dots,6}$	Drag Coefficients

References

- Shokirov, R.; Abdujabarov, N.; Jonibek, T.; Saytov, K.; Bobomurodov, S. Prospects of the Development of Unmanned Aerial Vehicles (UAVs). *Tech. Sci. Innov.* **2020**, *3*, 4–8. [\[CrossRef\]](#)
- Valavanis, K.; Vachtsevanos, G. *Handbook of Unmanned Aerial Vehicles*, 1st ed.; Springer: Dordrecht, The Netherlands, 2015; pp. 43–57.
- Bi, H.; Qi, G.; Hu, J. Modeling and Analysis of Chaos and Bifurcations for the Attitude System of a Quadrotor Unmanned Aerial Vehicle. *Complexity* **2019**, *2019*, 6313925. [\[CrossRef\]](#)
- Ziegler, J.G.; Nichols, N.B.; Rochester, N.Y. Optimum settings for automatic controllers. *ASME* **1942**, *64*, 759–768. [\[CrossRef\]](#)
- Åström, K.J.; Häggglund, T. Revisiting the Ziegler-Nichols step response method for PID control. *J. Process Control* **2004**, *14*, 635–650. [\[CrossRef\]](#)
- Allu, N.; Toding, A. Tuning with Ziegler Nichols Method for Design PID Controller at Rotate Speed DC Motor. *IOP Conf. Ser. Mater. Sci. Eng* **2020**, *806*, 012046. [\[CrossRef\]](#)
- Liang, H.; Sang, Z.K.; Wu, Y.Z.; Zhang, Y.H.; Zhao, R. High Precision Temperature Control Performance of a PID Neural Network-Controlled Heater under Complex Outdoor Conditions. *Appl. Therm. Eng.* **2021**, *195*, 117234. [\[CrossRef\]](#)
- Yu, X.; Li, C.; Zhou, J. A constrained differential evolution algorithm to solve UAV path planning in disaster scenarios. *Knowl.-Based Syst.* **2020**, *204*, 106209. [\[CrossRef\]](#)
- Qu, C.; Gai, W.; Zhang, J.; Zhong, M. A Novel Hybrid Grey Wolf Optimizer Algorithm for Unmanned Aerial Vehicle (UAV) Path Planning. *Knowl.-Based Syst.* **2020**, *194*, 105530. [\[CrossRef\]](#)
- Kramer, O. *Genetic Algorithm Essentials*; Springer International Publishing AG: Berlin/Heidelberg, Germany, 2017; Volume 679, pp. 11–20.
- Hernandez-Romero, N.; Medina-Marin, J.; Seck-Tuoh-Mora, J.C. *Introducción a Matlab para Resolver Problemas de Ingeniería Aplicando Algoritmos Genéticos*; Instituto de Ciencias Básicas e Ingeniería, Universidad Autónoma del Estado de Hidalgo: Pachuca de Soto, Mexico, 2012; pp. 41–54.
- Subhadip, S. Genetic Algorithm: An Approach for Optimization (Using MATLAB). *Int. J. Latest Trends Eng. Technol. (IJLTET)* **2014**, *3*, 261–267.
- Tanabe, R.; Fukunaga, A. Success-history based parameter adaptation for Differential Evolution. In Proceedings of the 2013 IEEE Congress on Evolutionary Computation, CEC 2013, Cancun, Mexico, 20–23 June 2013; Volume 3, pp. 71–78.
- Valadi, J.; Siarry, P. *Applications of Metaheuristics in Process Engineering*, 1st ed.; Springer International Publishing: Cham, Switzerland; Paris, France, 2014; pp. 1–38.
- Sarker, R.; Kamruzzaman, J.; Newton, C. Evolutionary optimization (EvOpt): A brief review and analysis. *Int. J. Comput. Intell. Appl.* **2003**, *4*, 311–330. [\[CrossRef\]](#)
- Kumar, K.; Davim, J.P. *Optimization Using Evolutionary Algorithms and Metaheuristics: Applications in Engineering*; CRC Press: Boca Raton, FL, USA, 2020.
- Ustunkok, T.; Karakaya, M. Effect of PSO Tuned P, PD, and PID Controllers on the Stability of a Quadrotor. In Proceedings of the 2019 1st International Informatics and Software Engineering Conference (UBMYK), Ankara, Turkey, 6–7 November 2019; Volume 4, pp. 1–5.
- Erkol O. Attitude controller optimization of four-rotor unmanned air vehicle. *Int. J. Micro Air Veh.* **2017**, *10*, 42–49. [\[CrossRef\]](#)
- Sheta, A.; Braik, M.; Maddi, D.R.; Mahdy, A.; Aljahdali, S.; Turabieh, H. Optimization of Pid Controller to Stabilize Quadcopter Movements Using Meta-Heuristic Search Algorithms. *Appl. Sci.* **2021**, *11*, 6492. [\[CrossRef\]](#)
- Talbi, E. *Metaheuristics: From Design to Implementation*, 1st ed.; Wiley: Hoboken, NJ, USA, 2009; pp. 247–254.
- Mirjalili, S.; Mirjalili, S.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [\[CrossRef\]](#)
- Yang, Y.; Chen, H.; Heidari, A.A.; Gandomi, A.H. Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts. *Expert Syst. Appl.* **2021**, *177*, 114864. [\[CrossRef\]](#)
- Tanabe, R.; Fukunaga, A.S. Improving the Search Performance of SHADE Using Linear Population Size Reduction. In Proceedings of the 2014 IEEE Congress on Evolutionary Computation (CEC), Beijing, China, 6–11 July 2014; pp. 1658–1665.
- Piotrowski, A.P. L-SHADE Optimization Algorithms with Population-Wide Inertia. *Inf. Sci.* **2018**, *468*, 117–141. [\[CrossRef\]](#)
- Mohamed, A.W.; Hadi, A.A.; Fattouh, A.M.; Jambi, K.M. LSHADE with Semi-Parameter Adaptation Hybrid with CMA-ES for Solving CEC 2017 Benchmark Problems. In Proceedings of the 2017 IEEE Congress on Evolutionary Computation (CEC), Donostia, Spain, 5–8 June 2017; pp. 145–152.
- Faramarzi, A.; Heidarinejad, M.; Mirjalili, S.; Gandomi, A.H. Marine Predators Algorithm: A Nature-Inspired Metaheuristic. *Expert Syst. Appl.* **2020**, *152*, 113377. [\[CrossRef\]](#)

27. Li, S.; Chen, H.; Wang, M.; Heidari, A.A.; Mirjalili, S. Slime mould algorithm: A new method for stochastic optimization. *Future Gener. Comput. Syst.* **2020**, *111*, 300–323. [[CrossRef](#)]
28. Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [[CrossRef](#)]
29. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. [[CrossRef](#)]
30. Razinkova, A.; Kang, B.; Cho, H.; Jeon, H. Constant Altitude Flight Control for Quadrotor UAVs with Dynamic Feedforward Compensation. *Int. J. Fuzzy Log. Intell. Syst.* **2014**, *14*, 26–33. [[CrossRef](#)]