Precision enhancement in drone position control based on optimized PID: A comparative study

Imam Barket Ghiloubi[1], Latifa Abdou[1,2]
[1]Identification, Command, Control and Communication Laboratory LI3CUB, Mohamed Khider University, Biskra, Algeria
[2]Faculty of Technology, Mostefa Ben Boulaïd University, Batna, Algeria

Article Info

Article history:

Received Feb 16, 2026
Revised Apr 20, 2026
Accepted May 22, 2026

Keywords:

Flower pollination algorithm
Particle swarm optimization
Precision
Proportional integral derivative
Quadrotor
Trajectory tracking

ABSTRACT

This paper presents a comparative study of quadrotor trajectory tracking performance using proportional integral derivative (PID) controllers optimized by two nature-inspired metaheuristic algorithms: the flower pollination algorithm (FPA) and particle swarm optimization (PSO). A nonlinear dynamic model of the quadrotor is first established, capturing both translational and rotational motions. A cascaded PID control architecture is then designed, where the outer loop regulates position and the inner loop controls attitude and altitude. The PID gains are tuned offline by minimizing an objective function combining the integral of time-weighted absolute error (ITAE) and an overshoot penalization term. The optimized controllers are evaluated through three reference scenarios: step response, circular trajectory, and lemniscate trajectory. Quantitative performance metrics, including root mean square error (RMSE), mean absolute error (MAE), integral of absolute error (IAE), and ITAE, are used to assess tracking accuracy. Results show that the PID-FPA controller significantly outperforms PID-PSO in terms of overshoot reduction, faster settling time, smoother responses, and improved tracking precision, especially for complex trajectories. A robustness analysis is finally conducted by introducing a realistic synthetic wind disturbance modeled using sinusoidal components and stochastic turbulence. The FPA-tuned PID demonstrates strong robustness, maintaining accurate trajectory tracking under aerodynamic perturbations.

Corresponding Author:

Imam Barket Ghiloubi
Identification, Command, Control and Communication Laboratory LI3CUB, Mohamed Khider University
Biskra, Algeria
Email: imam.ghiloubi@univ-biskra.dz

1. INTRODUCTION

Quadrotors have attracted significant attention in both research and industrial communities due to their compact structure, high maneuverability, and vertical takeoff and landing capabilities, which enable their deployment in a wide range of applications such as aerial surveillance, environmental monitoring, inspection of complex infrastructures, search and rescue missions, and autonomous delivery services. The increasing demand for safe, accurate, and reliable autonomous flight in complex and dynamic environments has made trajectory tracking control a central challenge in quadrotor unmanned aerial vehicle (UAV) systems [1]. However, the highly nonlinear dynamics, strong coupling between translational and rotational motions, under-actuation, and susceptibility to external disturbances such as wind gusts and modeling uncertainties significantly complicate controller design [2]. As a result, the trajectory tracking problem of quadrotor UAVs has attracted significant research attention, with numerous nonlinear, adaptive, and robust control approaches being developed to ensure stability, high tracking accuracy, and robustness under practical operating conditions. These strategies are specifically designed to address the inherent system nonlinearities, modeling uncertainties, actuator constraints, and the presence of external disturbances. Among these approaches, backstepping-based control techniques have received significant attention due to their systematic design procedure and stability guarantees. Adaptive backstepping enhanced by neural networks and barrier Lyapunov functions is proposed in [3] to handle uncertainties, output constraints, and input saturation. A reduced-complexity adaptive neural backstepping scheme is further introduced in [4], where an intermediary control variable and scalar adaptive parameter maintain tracking accuracy while lowering computational burden. To ensure fast convergence and robustness, a finite-time adaptive integral backstepping fast terminal sliding mode controller is developed in [5]. Barrier Lyapunov–based backstepping and constrained backstepping control strategies are also investigated in [6] to guarantee stable trajectory tracking while respecting position and attitude constraints. A self-tuning backstepping controller for multirotor drones is proposed, using a neural network trained via metaheuristic-generated data to optimize energy, rise time, and overshoot, outperforming genetic algorithm (GA)-tuned proportional integral derivative (PID) [7].

In parallel, classical and adaptive PID-based controllers remain attractive due to their simplicity and ease of implementation. A robust adaptive PID-like controller with online gain adjustment, without requiring an accurate model, is proposed in [8]. Trajectory tracking control approach combining sliding mode control with an improved extended state observer (SMC-IESO) is proposed in [9] for quadrotor UAVs. The method employs a hierarchical position–attitude control structure and a disturbance observer to enhance tracking accuracy and disturbance rejection while reducing chattering. In [10], a gain-scheduled fault-tolerant PID control strategy was proposed for quadrotor UAVs to address actuator faults. The PID parameters were optimally tuned using a combination of genetic algorithm (GA) and artificial neural networks (ANN), enabling improved flight stability and trajectory tracking under rotor failure scenarios. Furthermore, a robust Type-2 fuzzy PID controller optimized via genetic algorithms is experimentally validated in outdoor windy conditions and shown to outperform conventional PID and backstepping controllers in [11]. To enhance robustness against disturbances and modeling uncertainties, sliding mode control (SMC)–based techniques have been widely explored. A backstepping sliding mode controller combined with a super-twisting observer is proposed in [12] to achieve accurate tracking under partial-state measurements. In [13], a backstepping–sliding mode controller enhanced with a super-twisting observer is proposed for quadrotor unmanned aerial vehicle (QUAVs), enabling real-time disturbance estimation, improved tracking, and reduced chattering under uncertainties. Fixed-time sliding mode control is investigated in [14], ensuring fast convergence independent of initial conditions. Fractional-order sliding mode control is designed in [15] to suppress payload oscillations in slung-load quadrotor systems. Adaptive non-singular terminal sliding mode control with disturbance observation is proposed in [16] to improve robustness and disturbance rejection. Hybrid and advanced SMC schemes are further developed in [17] to mitigate chattering effects, while formation and trajectory tracking using non-singular terminal super-twisting sliding mode control are addressed in [18]. Enhanced robustness and reduced control effort are achieved using fuzzy PID–based super-twisting sliding mode control in [19], and quaternion-based sliding mode control is proposed in [20] to eliminate Euler angle singularities under severe wind disturbances.

More recently, model predictive control (MPC) and learning-based control approaches have gained increasing attention due to their ability to explicitly handle constraints and optimize performance. Nonlinear MPC with contraction-based cost functions is developed in [21] to achieve smooth and stable trajectory tracking. A neural network-based parallel model predictive control (PMPC) approach is proposed [22] for quadrotor UAV trajectory tracking. The method reconstructs the UAV dynamics online using a parallel neural network model and experience replay to maintain prediction accuracy in changing environments. In [23], an interval type-2 fuzzy pre-compensation active disturbance rejection control (IT2FP-ADRC) method was proposed to improve the trajectory tracking performance of a quadrotor slung-load system under disturbances and uncertainties. The approach enhances disturbance estimation and system robustness using a modified extended state observer and fuzzy compensation. Efficient real-time NMPC implementations on resource-limited embedded platforms are demonstrated in [24] using quaternion-based modeling and C/GMRES optimization. A comprehensive review in [25] highlights MPC as a promising framework for trajectory tracking and dynamic obstacle avoidance in complex UAV environments. In [26], a systematic linear quadratic regulator (LQR) tuning approach for UAV attitude control is proposed, using multi-objective grey wolf optimization and TOPSIS to select optimal Q–R weights under various error criteria. The study presented in [27] proposes a state-feedback controller with integral action for 3DOF quadrotor hover, achieving error-free, robust, and fault-tolerant attitude tracking in simulations and experiments. A model-free FPID controller is proposed for 3-DOF quadrotor attitude stabilization, optimized via particle swarm optimization (PSO) and validated against LQR, PID, and backstepping with robust performance [28]. In addition to individual quadrotor control, cooperative control strategies have also been explored. A fractional-order PID controller is presented in [29] to enhance QUAVs trajectory tracking, reduce overshoot, improve robustness, and enable scalable swarm operations in complex missions.

Despite the various advanced control strategies proposed in the literature, PID controllers remain attractive because of their simplicity, ease of implementation, and practical effectiveness. However, their performance heavily depends on the tuning of controller gains. Conventional tuning methods are often insufficient for nonlinear systems like quadrotors, motivating the use of intelligent optimization algorithms.

This paper focuses on optimizing PID controller gains using the particle swarm optimization (PSO) and the flower pollination algorithm (FPA). The main contributions of this study are summarized as follows:

- Unlike conventional ITAE-based tuning, an enhanced objective function is proposed by combining the integral of time-weighted absolute error (ITAE) with an overshoot penalization term to improve transient response and reduce overshoot, clearly distinguishing this work from our previous studies.
- Comprehensive comparative analysis between FPA-based and PSO-based PID controllers using multiple scenarios, including step response, circular, and lemniscate trajectories.
- Quantitative evaluation of tracking performance using different error metrics such as root mean square error (RMSE), mean absolute error (MAE), integral of absolute error (IAE), and ITAE.
- Validation on complex reference trajectories, including lemniscate paths, is performed to assess controller performance under more demanding tracking conditions than conventional step or circular trajectories.
- Robustness test against wind disturbances is conducted on the PID controller optimized using the selected method.

The remainder of this paper is organized as follows: Section 2 presents the quadrotor modeling, section 3 describes the control methodology, including the PID structure and optimization strategies, section 4 discusses the simulation results and section 5 presents the conclusion.

2. QUADROTOR MODELING

The dynamic model of the quadrotor captures the translational motion along the three axes (x, y, z) and the rotational dynamics around roll, pitch, and yaw. The aerodynamic forces and torques generated by the four rotors are mapped into the translational and angular accelerations of the quadrotor. To simplify the modeling for controller design, the following assumptions are made: The quadrotor is considered as a rigid body with symmetric structure and aerodynamic drag forces are neglected compared to thrust forces.

Figure 1 illustrates the quadrotor structure together with its inertial and body-fixed reference frames. The inertial frame (x, y, z) is used to describe the translational motion of the vehicle, while the body frame is attached to the quadrotor and defines its rotational dynamics. The attitude of the quadrotor is represented by the roll (ϕ), pitch (θ), and yaw (ψ) angles. The four rotors rotate with angular speeds $[\omega 1 \ldots \omega 4]$, generating thrust forces and control torques that allow independent control of altitude, attitude, and horizontal motion.

Mathematically, the quadrotor can be modeled by the following nonlinear state space model [7]:

$$
\begin{cases}
\dot{x}_1 = x_2 \\
\dot{x}_2 = \frac{I_y - I_z}{I_x} x_4 x_6 + \frac{L}{I_x} U_\phi \\
\dot{x}_3 = x_4 \\
\dot{x}_4 = \frac{I_z - I_x}{I_y} x_2 x_6 + \frac{L}{I_y} U_\theta \\
\dot{x}_5 = x_6 \\
\dot{x}_6 = \frac{I_x - I_y}{I_z} x_2 x_4 + \frac{L}{I_z} U_\psi \\
\dot{x}_7 = x_8 \\
\dot{x}_8 = \frac{\cos x_1 . \sin x_3 . \cos x_5 + \sin x_1 . \sin x_5}{m} U_z - \frac{K_x x_8}{m} + \frac{F_x}{m} \\
\dot{x}_9 = x_{10} \\
\dot{x}_{10} = \frac{\cos x_1 . \sin x_3 . \sin x_5 - \sin x_1 . \cos x_5}{m} U_z - \frac{K_y x_{10}}{m} + \frac{F_y}{m} \\
\dot{x}_{11} = x_{12} \\
\dot{x}_{12} = \frac{\cos x_1 . \cos x_3}{m} U_z - \frac{K_z x_{12}}{m} - g + \frac{F_z}{m}
\end{cases}
\tag{1}
$$

where

$$\begin{cases} x_1 = \phi \\ x_2 = \dot{\phi} \\ x_3 = \theta \\ x_4 = \dot{\theta} \\ x_5 = \psi \\ x_6 = \dot{\psi} \\ x_7 = x \\ x_8 = \dot{x} \\ x_9 = y \\ x_{10} = \dot{y} \\ x_{11} = z \\ x_{12} = \dot{z} \end{cases} \tag{2}$$

and F_x, F_y, F_z are the aerodynamic forces generated by wind disturbances along each axis, given by (3):

$$\left\{ F_i = \frac{1}{2}\sigma C_d A V_i^2 \quad , \ i = x, y, z \right. \tag{3}$$

and applied in the last section of results to evaluate the robustness of the selected optimized controller PID.

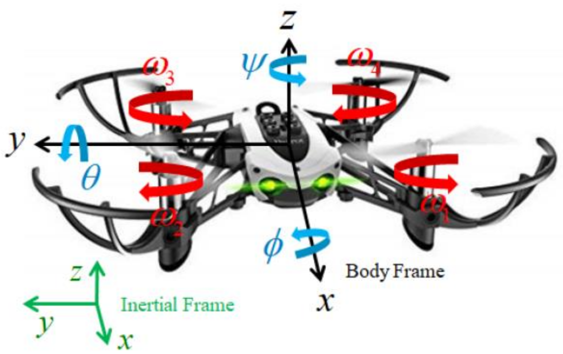


Figure 1. Quadrotor reference frames and degrees of freedom

The dynamic model of the quadrotor used in this work incorporates linear aerodynamic drag forces along each body axis, represented by K_x, K_y and K_z, to maintain physical consistency. External disturbances due to wind, denoted by F_x, F_y and F_z are also explicitly considered in the simulations to evaluate robustness under realistic operating conditions. While these effects are included in the model for completeness, the control design focuses on achieving precise trajectory tracking and disturbance rejection without directly compensating for the drag, allowing a clear assessment of the controller's performance under both ideal and perturbed scenarios.

The parameters of the quadrotor used in the dynamic model, including the mass, geometric dimensions, gravitational constant, and moments of inertia, are summarized in Table 1. Numerical simulations are carried out using a fourth-order Runge–Kutta integration scheme with a fixed step size of 0.01 s, which provides a good balance between computational efficiency and accuracy.

Table 1. Physical parameters of the quadrotor model

Parameter	Description	Value
m	Total mass	1 kg
L	Distance from the center of mass to each rotor	0.25 m
g	Gravitational acceleration	9.81 m/s^2
I_x	Moment of inertia around the body x-axis (roll axis)	0.02 $kg.m^2$
I_y	Moment of inertia around the body y-axis (pitch axis)	0.02 $kg.m^2$
I_z	Moment of inertia around the body z-axis (yaw axis)	0.04 $kg.m^2$
K_x	aerodynamic drag coefficient along the x-axis	0.1 $N.s/m$
K_y	aerodynamic drag coefficient along the y-axis	0.1 $N.s/m$
K_z	aerodynamic drag coefficient along the z-axis	0.2 $N.s/m$

3. CONTROL METHOD
3.1. PID control structure
The control architecture is based on a cascaded PID scheme. The outer loop regulates the position (x, y), while the inner loop controls the attitude angles (ϕ, θ, ψ) and altitude z as shown in Figure 2.

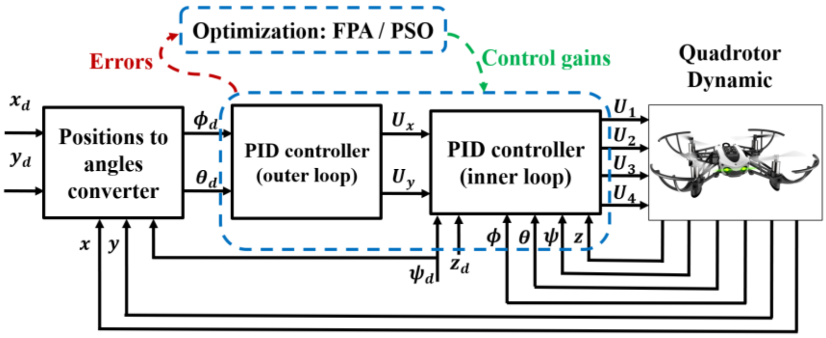


Figure 2. Diagram of the optimized PID control architecture

The desired position references (x_d, y_d, z_d) are processed by an outer-loop PID controller that generates intermediate virtual control signals U_x, U_y, and U_z based on the corresponding position errors. The horizontal virtual control signals U_x and U_y are then converted into desired roll and pitch angles ϕ_d and θ_d through the following relations:

$$\begin{cases} \phi_d = \frac{1}{g}(U_y cos\psi - U_x sin\psi) \\ \theta_d = \frac{1}{g}(U_x cos\psi + U_y sin\psi) \end{cases} \tag{4}$$

An outer-loop PID controller generates the virtual control inputs U_x and U_y using the tracking errors in position, while an inner-loop PID controller regulates the attitude (ϕ, θ, ψ) based on the references $(\phi_d, \theta_d, \psi_d)$. The PID gains of both loops are optimized offline using the FPA and PSO algorithms. The optimized control inputs $(U_z, U_\phi, U_\theta, U_\psi)$ are then applied to the quadrotor dynamic model, and the measured states are fed back to close the control loop.

Based on the control structure illustrated in Figure 2, the quadrotor motion along six degrees of freedom - three translational (x, y, z) and three rotational (ϕ, θ, ψ)- is regulated using a unified PID control law. For each controlled variable x, y, z, ϕ, θ and ψ, the corresponding control input U_j is defined as:

$$U_j = Kp_j e_j + Ki_j \int_0^t e_j dt + Kd_j \dot{e}_j , \quad j \in \{x, y, z, \phi, \theta, \psi\} \tag{5}$$

where $e_j = j_d - j$ denotes the tracking error between the desired and actual states. The proportional, integral, and derivative gains Kp_j, Ki_j, Kd_j are tuned independently for each degree of freedom using the FPA and PSO optimization algorithms.

For the translational dynamics, the virtual control inputs U_x and U_y are converted into the desired roll and pitch angles (ϕ_d, θ_d), while U_z directly regulates the altitude. For the rotational dynamics: U_ϕ, U_θ, U_ψ correspond directly to the roll, pitch, and yaw control torques applied to the quadrotor.

The PID gains are tuned by minimizing the following objective function:

$$J = \int_0^T t(|e_x| + |e_y| + |e_z|) + \alpha(OS_x + OS_y + OS_z)dt \tag{6}$$

where e_x, e_y and e_z denote the instantaneous position errors along each axis, computed as the difference between the desired and actual UAV positions. OS_x , OS_y and OS_z represent the maximum overshoot along each axis, defined as the largest deviation of the UAV position from the reference trajectory during the simulation or experimental run. The integration is performed over the fixed time interval T, corresponding to the duration of the tested trajectory, ensuring consistency and reproducibility of the results. The weighting factor α balances the trade-off between minimizing overshoot and improving tracking accuracy.

3.2. PID based on FPA optimization

The flower pollination algorithm (FPA) is a metaheuristic optimization technique based on the pollination process of flowering plants. In the proposed approach, each candidate solution is represented by a decision vector:

$$x_j = [kp \quad ki \quad kd]_j \tag{7}$$

which contains the proportional, integral, and derivative gains of the PID controller. The objective of the optimization process is to determine the optimal PID parameters that minimize the performance index J. FPA alternates between global and local pollination mechanisms. Global pollination models cross-pollination and is performed using Levy flights, which enable long-distance exploration toward the current best solution g^*. The global pollination update rule is given by:

$$x_i^{t+1} = x_i^{\,t} + L(x_i^{\,t} - g^*) \tag{8}$$

where $x_i^{\,t}$ denotes the PID gain vector of the i^{th} candidate solution at iteration t, g^* represents the best PID gain vector found so far, and L is the Levy flight step size following a heavy-tailed probability distribution as shown in Table 2.

Table 2. Optimization parameters for PSO and FPA algorithms

Parameter	FPA	PSO
Population size	200	200
Number of iterations	300	300
Global pollination probability p	0.8	/
Lévy flight step L	1.5	/
Acceleration coefficient c_1	/	1.7
Acceleration coefficient c_2	/	1.5
Inertia weight w	/	0.7
Search space gain bounds for $[kp \quad ki \quad kd]$: (Lower bounds: [0, 0, 0]; upper bounds as specified below)		
Position x, y	[8, 4, 3]	[8, 4, 3]
Position z	[12, 4, 5]	[12, 4, 5]
Angles φ, θ	[3, 2, 3]	[3, 2, 3]
Angle ψ	[4, 2, 2]	[4, 2, 2]

Local pollination represents self-pollination within a local neighborhood, allowing fine-tuning of the PID gains by exchanging information between randomly selected candidate solutions. A switching probability p controls the transition between global and local pollination. At each iteration, the fitness of each candidate solution is evaluated using the objective function J, which combines the ITAE and an overshoot penalty. Over successive generations, the FPA converges toward the optimal set of PID gains that minimizes J [30].

3.3. PID based on PSO optimization

Particle swarm optimization (PSO) is a population-based stochastic optimization algorithm inspired by the collective behavior of bird flocks and fish schools. Similar to the FPA-based approach described in the previous section, each particle is associated with a PID gain vector defined in (7), which represents a candidate solution in the search space.

The dynamic evolution of each particle is governed by its velocity and position update equations:

$$\begin{cases} v^{t+1} = wv_i^{\,t} + c_1 r_1 (p_i - x_i^{\,t}) + c_2 r_2 (g - x_i^{\,t}) \\ \qquad x_i^{t+1} = x_i^{\,t} + v_i^{t+1} \end{cases} \tag{9}$$

where $x_i^{\,t}$ and $v_i^{\,t}$ denote the position (PID gains) and velocity of the i^{th} particle at iteration t, respectively. The term p_i represents the personal best PID gain vector achieved by the particle, while g denotes the global best solution found by the swarm. The parameters w, c_1, and c_2 are the inertia weight and acceleration coefficients controlling the trade-off between exploration and exploitation, and r_1, r_2 are random numbers uniformly distributed in [0,1] as shown in Table 2.

Through iterative updates, particles move toward regions of the search space associated with lower cost values. The optimization process continues until convergence criteria are met, resulting in an optimal or near-optimal set of PID gains that minimizes the objective function J [30].

In this work, FPA and PSO are employed as optimization techniques to determine suitable PID controller gains that minimize the defined objective function. These metaheuristic algorithms do not modify the structure of the control law but instead search for appropriate parameter values within the predefined bounds. Consequently, the stability of the closed-loop system depends on the PID controller design and the quadrotor dynamic model, while the optimization algorithms act solely as tools for parameter tuning. All simulations start from zero initial conditions for position, velocity, and attitude states. The main parameters used for the PID optimization with the FPA and PSO are summarized in Table 2.

Table 2 provides detailed settings for both optimization algorithms, including PID gain search space, population size, number of iterations, and algorithm-specific parameters such as pollination probability, Levy exponent for FPA, and inertia weight and acceleration coefficients for PSO. These settings ensure reproducibility of the study and a fair comparison between the two approaches.

4. RESULTS AND DISCUSSION

To evaluate the performance of the optimized PID controllers, three reference scenarios: Step response, circular and lemniscate trajectories were simulated, followed by a robustness test of the best-performing controller.

4.1. First scenario: Step response

A step test is carried out with desired values of:

$$\begin{cases} x_d = 1\,m \\ y_d = 0.5\,m \\ z_d = 1\,m \end{cases} \tag{10}$$

Figure 3 illustrates the step response performance of the quadrotor under both PID-FPA and PID-PSO controllers. The time-responses for the x, y, and z positions clearly show that the PID-PSO exhibits noticeable overshoot in all three axes, while the PID-FPA effectively eliminates overshoot. Moreover, the PID-FPA achieves faster convergence, particularly in the x and z responses, indicating superior transient behavior as shown in Figure 3(a). The 3D trajectory in Figure 3(b) representation further confirms these observations, highlighting the ability of the PID-FPA to provide more accurate and stable tracking performance compared to the PID-PSO.

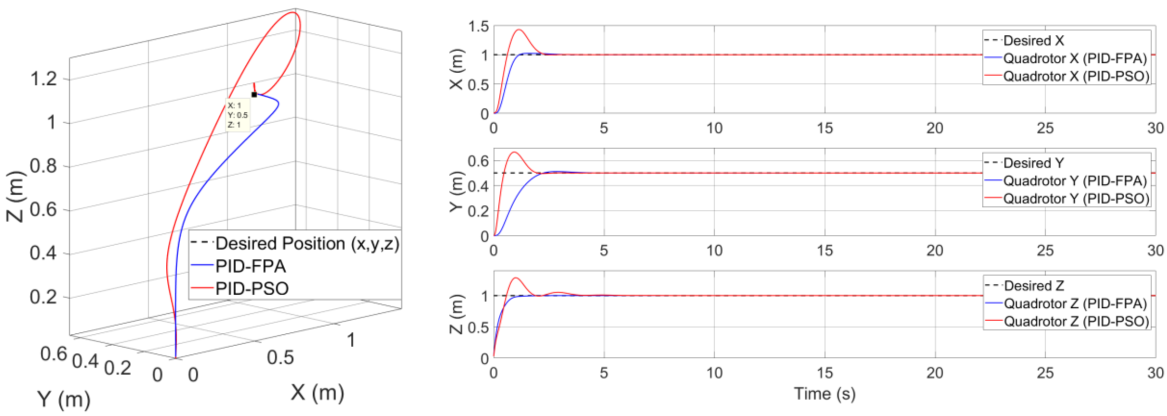


Figure 3. Step response: (a) 3D motion and (b) x, y, z positions versus time

Table 3 compares the performance of the quadrotor system under PID controllers optimized by PSO and FPA algorithms. Both methods achieve a zero steady-state error, indicating accurate tracking in steady state. However, clear differences appear in the transient response.

For the x-axis, the PID-FPA controller achieves a faster settling time compared with PID-PSO and significantly reduces the overshoot from 43% to 2.3%. Along the y-axis, both controllers exhibit similar settling times, but the FPA-based controller provides a much smaller overshoot. For the z-axis, the improvement is the most remarkable, with the settling time reduced from 3.6 to 1 s and the overshoot completely eliminated.

Table 3. Performance comparison of PID controllers tuned via PSO and FPA optimization

Axe	Controller	Settling Time (s)	Overshoot (%)
Position "x"	PID-PSO	2.2	43
	PID-FPA	1.1	2.3
Position "y"	PID-PSO	1.8	33.3
	PID-FPA	2	2.2
Altitude "z"	PID-PSO	3.6	28.6
	PID-FPA	1	0

4.2. Second scenario: Circular trajectory

In this case, a circular trajectory with a radius of 1 m and altitude of 1 m is considered:

$$\begin{cases} x_d = 1 \cos 0.5t \ m \\ y_d = 1 \sin 0.5t \ m \\ z_d = 1 \ m \end{cases} \tag{11}$$

Figure 4 presents the quadrotor performance when tracking a circular trajectory. Although both PID-FPA and PID-PSO approaches eventually converge to the desired trajectory in steady state, the PID-FPA demonstrates superior stability and accuracy during the initial tracking phase. In contrast, the PID-PSO in Figure 4(a) shows less precise convergence at the start. These differences are more evident in the 3D trajectory plot, shown in Figure 4(b) where the PID-FPA maintains a smoother and closer path to the reference circle compared to the PID-PSO.

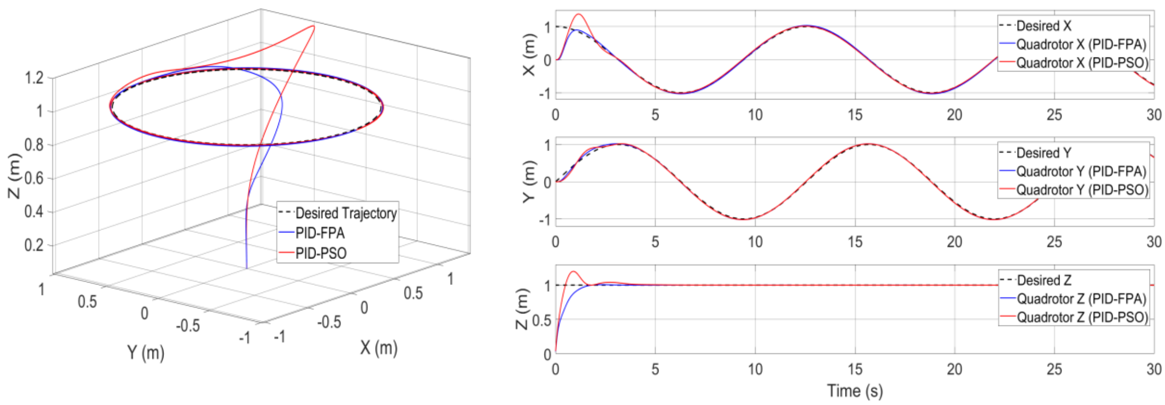


Figure 4. Circular trajectory tracking: (a) 3D motion and (b) x, y, z positions versus time

4.3. Third scenario: Lemniscate (∞) trajectory

To further evaluate performance, a lemniscate trajectory with a radius of 1 m and altitude of 1 m is applied as a desired trajectory:

$$\begin{cases} x_d = \frac{1 \cos 0.25t}{1+\sin^2(0.25t)} \ m \\ y_d = \frac{1 \cos(0.25t)\sin(0.25t)}{1+\sin^2(0.25t)} \ m \\ z_d = 1 \ m \end{cases} \tag{12}$$

Figure 5 shows the quadrotor tracking performance for a lemniscate (∞-shaped) trajectory of 1 m radius at 1 m altitude. This trajectory is more complex and requires higher tracking accuracy compared to the circular path. The results highlight that the PID-FPA provides smoother and more stable tracking along the entire trajectory, maintaining closer adherence to the desired path, as shown in Figure 5(a). In contrast, the PID-PSO exhibits noticeable deviations, particularly in the curved sections, confirming the superior robustness and precision of the PID-FPA in handling complex maneuvers as shown in Figure 5(b).

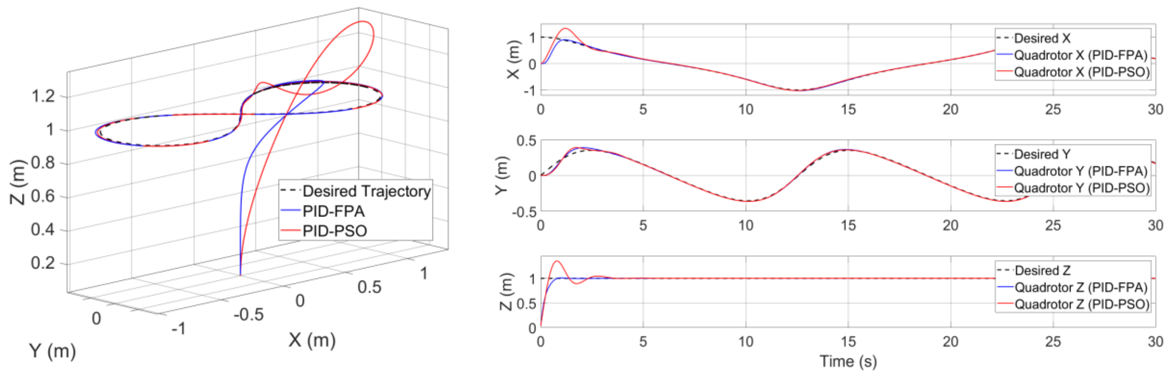


Figure 5. Lemniscate trajectory tracking: (a) 3D motion and (b) x, y, z positions versus time

4.4. Quantitative comparison of trajectory tracking performance

In order to provide an objective and quantitative evaluation of the trajectory tracking performance, additional error-based metrics are considered to complement the time-domain responses and qualitative plots presented in the previous subsections. These metrics allow a direct comparison between the PSO-optimized PID and the FPA-optimized PID controllers under identical simulation conditions.

Let the position tracking error vector be defined as:

$$\begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix} = \begin{bmatrix} x_d(t) - x(t) \\ y_d(t) - y(t) \\ z_d(t) - z(t) \end{bmatrix} \tag{13}$$

where (x_d, y_d, z_d) and (x, y, z) denote the desired and actual quadrotor positions, respectively.

The following quantitative performance indices are employed to evaluate the 3D trajectory tracking (x, y, z) performance of the proposed controller.

– Root mean square error (RMSE):

$$RMSE = \sqrt{\frac{1}{T}\int_0^T e(t)^2 dt} \tag{14}$$

which measures the overall tracking accuracy over the entire trajectory and is sensitive to large transient errors.

– Mean absolute error (MAE):

$$MAE = \frac{1}{T}\int_0^T t|e(t)|dt \tag{15}$$

which provides an average measure of the absolute tracking error over the full simulation horizon.

– Integral of absolute error (IAE):

$$IAE = \int_0^T |e(t)|dt \tag{16}$$

which quantifies the cumulative tracking error and reflects the overall deviation from the reference trajectory.

– Integral of time-weighted absolute error (ITAE):

$$ITAE = \int_0^T t|e(t)|dt \tag{17}$$

which penalizes persistent tracking errors occurring at later time instants and is consistent with the objective function used during the PID tuning process.

where: $e(t) = \sqrt{e_x{}^2 + e_y{}^2 + e_z{}^2}$ denotes the Euclidean norm of the 3D position tracking error, and T represents the total simulation time with $t \in [0, T]$.

All metrics are computed for each trajectory scenario and summarized in Table 4. The table presents the quantitative tracking performance metrics obtained for the circular and lemniscate trajectories using the PID controllers tuned via FPA and PSO. For both trajectories, the PID-FPA controller consistently achieves lower RMSE and MAE values compared to PID-PSO, indicating improved overall tracking accuracy and reduced average error magnitude. In particular, for the lemniscate trajectory, PID-FPA exhibits a noticeable reduction in RMSE (0.137204 vs 0.147136) and MAE (0.032794 vs 0.043116), highlighting its superior ability to handle more complex and highly nonlinear reference paths. Furthermore, the integral performance indices (IAE and ITAE) reported in Table 4 confirm the advantage of the FPA-based tuning strategy. The PID-FPA controller yields lower IAE values for both circular and lemniscate trajectories, reflecting reduced cumulative tracking error over the full simulation horizon. More importantly, the ITAE metric, which penalizes persistent errors at later time instants and is consistent with the optimization objective, is significantly smaller for PID-FPA, particularly in the lemniscate case (5.363847 vs 6.183683). This indicates faster error attenuation and improved steady-state tracking performance. Overall, the results demonstrate that the FPA-based PID controller provides more accurate and efficient 3D trajectory tracking than its PSO-based counterpart, especially for trajectories with higher curvature and dynamic complexity. Overall, the PID-FPA controller achieves faster convergence, smoother responses, and improved damping characteristics compared to PID-PSO, which is consistent with the superior quantitative tracking metrics and confirms the higher efficiency of the flower pollination algorithm for PID gain optimization.

Table 4. Quantitative tracking performance metrics

Method	Trajectory	RMSE	MAE	IAE	ITAE
PID-FPA	Circular	0.143865	0.050983	1.522934	12.700344
	Lemniscate	0.137204	0.032794	0.977154	5.363847
PID-PSO	Circular	0.149294	0.058172	1.738653	12.937651
	Lemniscate	0.147136	0.043116	1.286909	6.183683

From a computational perspective, both PSO and FPA belong to population-based metaheuristic optimization algorithms. PSO updates particle velocities and positions using local and global best solutions, while FPA relies on global pollination based on Lévy flights and local pollination mechanisms. Although both algorithms have comparable computational complexity depending on population size and iteration number, FPA often requires fewer iterations to converge toward high-quality solutions due to its balance between global exploration and local exploitation.

4.5. Robustness test under wind disturbances

A wind profile is introduced as an external disturbance to assess the robustness of the quadrotor control system. In accordance with the results of the previous subsection, where the FPA-tuned PID controller was selected due to its superior performance compared to the PSO-based approach, the robustness analysis is conducted using the FPA-PID controller. The objective is to evaluate its capability to maintain accurate trajectory tracking under environmental perturbations.

The wind velocity components along the horizontal axes are defined as:

$$\begin{cases} V_x(t) = [1 + sin(0.7t) + 0.5sin(0.5t)] + X_x(t) \\ V_y(t) = [1 + 1.2sin(0.7t + \pi/3)] + X_y(t) \end{cases} \tag{18}$$

The turbulent components evolve according to the following stochastic difference equations:

$$\begin{cases} X_x(t) = \alpha_x X_x(t-\Delta t) + \sigma_x\sqrt{1 - {\alpha_x}^2}\eta_x(t) \,, \alpha_x = e^{\frac{-\Delta t}{\tau_x}} \\ X_y(t) = \alpha_y X_y(t-\Delta t) + \sigma_y\sqrt{1 - {\alpha_y}^2}\eta_y(t) \,, \alpha_y = e^{\frac{-\Delta t}{\tau_y}} \end{cases} \tag{19}$$

The effect of wind on the quadrotor is then modeled through aerodynamic drag forces acting along each axis, expressed using the classical quadratic drag formulation:

$$\begin{cases} F_x^w = \frac{1}{2}\sigma C_d A {V_x}^2 \\ F_y^w = \frac{1}{2}\sigma C_d A {V_y}^2 \end{cases} \tag{20}$$

where: $\rho = 1.225 \frac{kg}{m^3}$ is the air density, $Cd = 1$ is a typical drag coefficient, $A = 0.05\, m^2$: represents the estimated frontal area of the $1kg$ quadrotor.

The synthetic wind profile is modelled as the sum of a low-frequency mean component (sinusoids) and a correlated stochastic turbulence of the Ornstein-Uhlenbeck type. The parameters are chosen to represent moderate wind conditions realistic enough to assess the drone's robustness against atmospheric disturbances. The turbulence standard deviation is set to $\sigma_x = \sigma_y = 1.2\, m$ on the horizontal axes and $\sigma_z = 0.5\, m$ on the vertical axis, with a common correlation time $\tau = 4\, s$ for all three axes. The deterministic component includes a residual mean wind of $1\, m/s$ along x and y, plus sinusoidal oscillations: on the x-axis, an amplitude of $1\, m/s$ at $0.7\, rad/s$ and a second amplitude of $0.5\, m/s$ at $0.5\, rad/s$ on the y-axis, an amplitude of $1.2\, m/s$ at $0.7\, rad/s$ with a phase shift $\pi/3$ on the z-axis, an amplitude of $1\, m/s$ at $0.7\, rad/s$. Although these values do not stem from a classical logarithmic profile, they generate gusts and directional variations consistent with flight scenarios in perturbed environments, and are suitable for Simulink simulation with a time step of $0.01\, s$ [31]. This disturbance model is applied to both circular and lemniscate trajectory tracking cases to thoroughly evaluate the controller's robustness.

4.5.1. First scenario: Circular trajectory

The first robustness assessment is performed for the circular reference trajectory in the presence of the synthetic wind disturbance described previously. This scenario allows evaluating how the quadrotor reacts to continuous lateral aerodynamic forces while maintaining a smooth periodic motion. Since circular motion requires a constant coupling between the horizontal axes and altitude stabilization, it constitutes a demanding test when external perturbations are applied. The objective here is to analyze whether the FPA-PID controller can preserve trajectory accuracy despite the combined deterministic and stochastic wind components acting along the three axes.

Figure 6 illustrates the wind-induced aerodynamic forces applied to the quadrotor during the circular trajectory tracking. The forces exhibit non-stationary behavior resulting from the superposition of sinusoidal components and correlated stochastic turbulence. This produces irregular gusts and directional variations that continuously disturb the vehicle along the horizontal plane and, to a lesser extent, along the vertical axis. The fluctuating nature of these forces confirms that the quadrotor is subjected to realistic perturbations that challenge the stability and robustness of the control system throughout the motion.

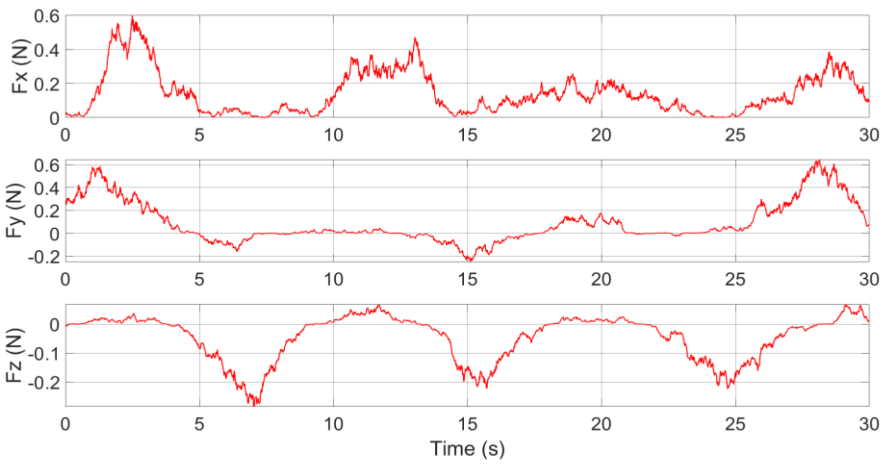


Figure 6. Wind profile forces applied to the quadrotor during circular trajectory tracking

Figure 7 presents the circular trajectory tracking performance under wind disturbance. The 3D representation shows that the quadrotor remains very close to the desired path despite the persistent external forces. The position plots along the x, y, and z axes as in Figure 7(a) indicate small deviations occurring mainly when wind gusts reach their peak values. However, these deviations are rapidly compensated by the controller, and no drift or instability is observed. As shown in Figure 7(b), this behavior highlights the strong disturbance rejection capability of the FPA-PID controller, which maintains accurate tracking even under continuously varying aerodynamic loads.

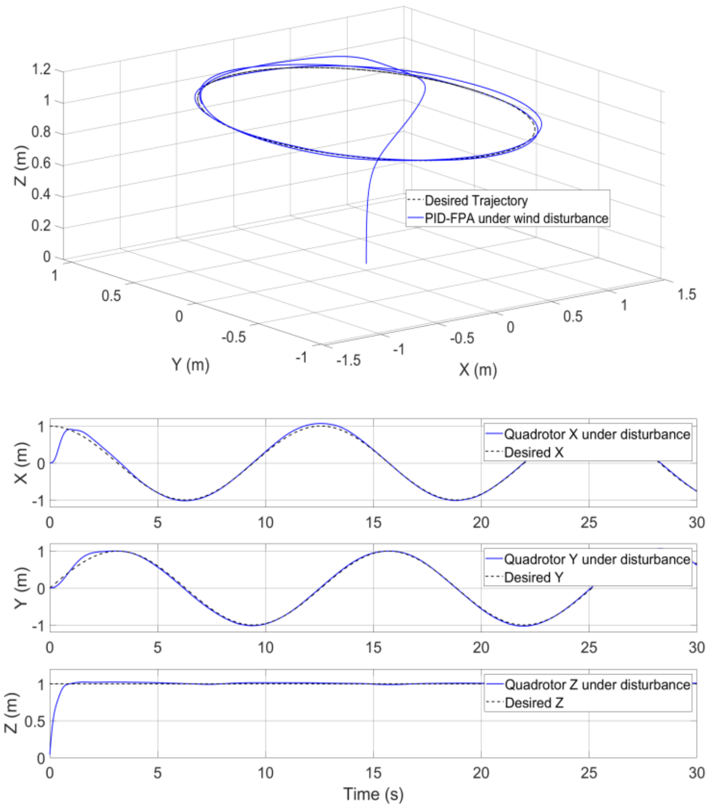


Figure 7. Circular trajectory tracking under wind disturbance: (a) 3D motion and (b) x, y, z positions

4.5.2. Second scenario: Lemniscate (∞) trajectory

The second robustness evaluation considers the lemniscate (∞) trajectory, which is more challenging than the circular path due to its sharp curvature changes and frequent reversals in direction. This trajectory imposes rapid variations in velocity and acceleration, making the quadrotor more sensitive to external disturbances. Testing the controller under this configuration allows assessing its ability to handle wind perturbations during aggressive maneuvering conditions and complex spatial motion.

Figure 8 shows the evolution of the wind-induced forces applied during the lemniscate tracking. Similar to the previous scenario, the forces present irregular oscillations caused by the combination of deterministic wind components and stochastic turbulence. Due to the frequent changes in direction imposed by the lemniscate path, the interaction between the aerodynamic forces and the quadrotor dynamics becomes more pronounced, generating transient disturbances that significantly challenge the tracking controller.

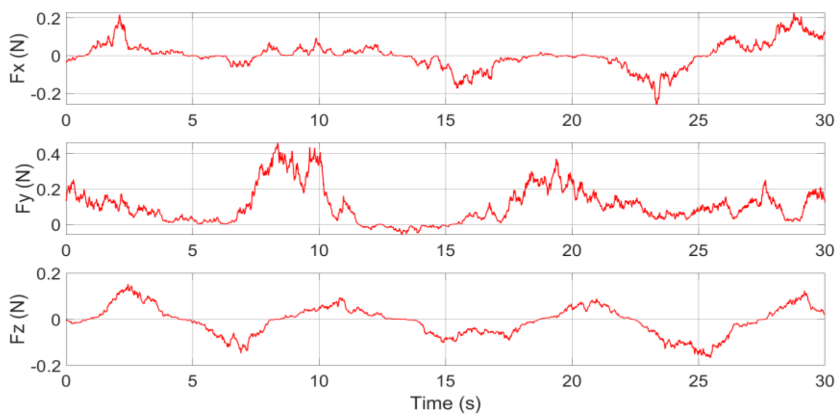
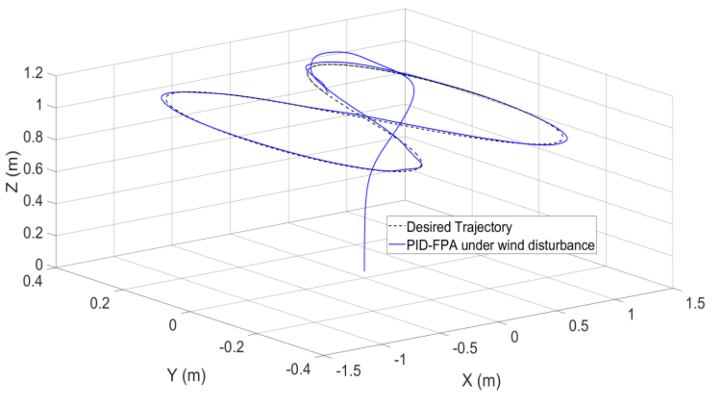


Figure 8. Wind profile forces applied to the quadrotor during lemniscate trajectory tracking

Figure 9 illustrates the lemniscate trajectory tracking under wind disturbance. The 3D trajectory plot in Figure 9(a) shows that, despite the increased complexity of the path and the presence of significant gusts, the quadrotor successfully follows the desired trajectory with limited tracking errors. Slight deviations are observed at the points in Figure 9(b), where the curvature changes rapidly, as the effect of wind adds to the dynamic demand of the maneuver. Nevertheless, the controller quickly stabilizes the motion without oscillations or loss of stability. This confirms that the FPA-PID controller ensures robust performance even for aggressive trajectories subjected to realistic atmospheric disturbances.



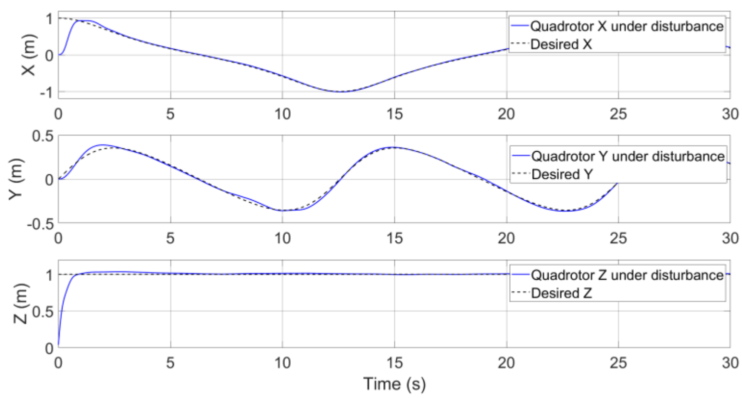


Figure 9. Lemniscate trajectory tracking under wind disturbance: (a) 3D motion (b) x, y, z positions

5. CONCLUSION

This study investigated the effectiveness of metaheuristic optimization techniques for tuning PID controllers applied to quadrotor trajectory tracking. A cascaded PID control structure was implemented on a nonlinear quadrotor model, and the flower pollination algorithm and particle swarm optimization were used to optimize the controller gains under an objective function based on ITAE and overshoot penalization. Simulation results obtained for step inputs, circular trajectories, and lemniscate paths clearly demonstrated that the PID controller tuned via FPA provides superior transient and steady-state performance compared to the PSO-based approach. The FPA-optimized controller achieved significantly lower overshoot, faster settling time, smoother responses, and improved accuracy in tracking complex trajectories. Quantitative error metrics (RMSE, MAE, IAE, ITAE) further confirmed the advantage of FPA, particularly for highly nonlinear reference paths. The robustness test performed under realistic wind disturbances showed that the selected PID-FPA controller maintains stable and accurate tracking despite persistent aerodynamic perturbations. This confirms the suitability of the proposed approach for quadrotor navigation in disturbed environments. Overall, the results highlight the effectiveness of the flower pollination algorithm as a powerful and efficient tool for PID gain optimization in UAV control applications. Experimental validation and extension toward more sophisticated control strategies will be the main focus of future work.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Imam Barket Ghiloubi	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓			
Latifa Abdou	✓			✓	✓					✓	✓	✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The authors confirm that the data supporting the findings of this study are available within the article [and/or its supplementary materials].

REFERENCES

[1] M. M. Madebo, C. M. Abdissa, L. N. Lemma, and D. S. Negash, "Robust tracking control for quadrotor UAV with external disturbances and uncertainties using neural network based MRAC," *IEEE Access*, vol. 12, pp. 36183–36201, 2024, doi: 10.1109/ACCESS.2024.3374894.

[2] E. A. Gedefaw, N. B. Abera, and C. M. Abdissa, "A review of modeling and control techniques for unmanned aerial vehicles," *Engineering Reports*, vol. 7, no. 6, p. e70215, Jun. 2025, doi: 10.1002/eng2.70215.

[3] J. Li, L. Wan, J. Li, and K. Hou, "Adaptive backstepping control of quadrotor UAVs with output constraints and input saturation," *Applied Sciences*, vol. 13, no. 15, p. 8710, Jul. 2023, doi: 10.3390/app13158710.

[4] X. Song, L. Shen, and F. Chen, "Adaptive backstepping position tracking control of quadrotor unmanned aerial vehicle system," *AIMS Mathematics*, vol. 8, no. 7, pp. 16191–16207, 2023, doi: 10.3934/math.2023828.

[5] K. Eliker and W. Zhang, "Finite-time adaptive integral backstepping fast terminal sliding mode control application on quadrotor UAV," *International Journal of Control, Automation and Systems*, vol. 18, no. 2, pp. 415–430, Feb. 2020, doi: 10.1007/s12555-019-0116-3.

[6] A. Khadhraoui, A. Zouaoui, and M. Saad, "Barrier Lyapunov function and adaptive backstepping-based control of a quadrotor UAV," *Robotica*, vol. 41, no. 10, pp. 2941–2963, Oct. 2023, doi: 10.1017/S0263574723000735.

[7] O. Rodríguez-Abreo, M. Aviles, J. Rodríguez-Reséndiz, and A. García-Cerezo, "Neural networks and genetic algorithms-based self-adjustment system for a backstepping controller of an unmanned aerial vehicle," *Alexandria Engineering Journal*, vol. 126, pp. 70–80, Jul. 2025, doi: 10.1016/j.aej.2025.04.034.

[8] A. Boubakir, T. Souanef, S. Labiod, and J. F. Whidborne, "A robust adaptive PID-like controller for quadrotor unmanned aerial vehicle systems," *Aerospace*, vol. 11, no. 12, p. 980, Nov. 2024, doi: 10.3390/aerospace11120980.

[9] X. Wu, Y. Pan, Q. Chen, N. Zheng, and Z. Chen, "Attitude control of a quadcopter UAV using sliding mode control with an improved extended state observer," *Electronics*, vol. 14, no. 22, p. 4416, Nov. 2025, doi: 10.3390/electronics14224416.

[10] K. Surur, I. Kabir, G. Ahmad, and M. A. Abido, "Optimal gain scheduling for fault-tolerant control of quadrotor UAV using genetic algorithm-based neural network," *Arabian Journal for Science and Engineering*, pp. 1–15, Mar. 2025, doi: 10.1007/s13369-025-09994-y.

[11] O. Moali, D. Mezghani, A. Mami, A. Oussar, and A. Nemra, "UAV trajectory tracking using proportional-integral-derivative-type-2 fuzzy logic controller with genetic algorithm parameter tuning," *Sensors*, vol. 24, no. 20, p. 6678, Oct. 2024, doi: 10.3390/s24206678.

[12] V. Borja-Jaimes, J. García-Morales, R. F. Escobar-Jiménez, G. V. Guerrero-Ramírez, and M. Adam-Medina, "A backstepping sliding mode control of a quadrotor UAV using a super-twisting observer," *Applied Sciences*, vol. 15, no. 18, p. 10120, Sep. 2025, doi: 10.3390/app151810120.

[13] V. Borja-Jaimes, J. S. Valdez-Martínez, M. Beltrán-Escobar, G. Ramírez-Zúñiga, A. Reyes-Mayer, and M. Calixto-Rodríguez, "Robust Backstepping-sliding control of a quadrotor UAV with disturbance compensation," *Computation*, vol. 14, no. 2, p. 51, Feb. 2026, doi: 10.3390/computation14020051.

[14] J. Olguin-Roque, S. Salazar, I. González-Hernandez, and R. Lozano, "A robust fixed-time sliding mode control for quadrotor UAV," *Algorithms*, vol. 16, no. 5, p. 229, Apr. 2023, doi: 10.3390/a16050229.

[15] M. Al-Dhaifallah, F. M. Al-Qahtani, S. Elferik, and A.-W. A. Saif, "Quadrotor robust fractional-order sliding mode control in unmanned aerial vehicles for eliminating external disturbances," *Aerospace*, vol. 10, no. 8, p. 665, Jul. 2023, doi: 10.3390/aerospace10080665.

[16] H. Zhang, F. Zeng, F. Huang, and S. Zhang, "Quadrotor UAV trajectory tracking control based on adaptive non-singular terminal sliding mode," *Journal of Electrical Systems*, vol. 20, no. 3, 2024.

[17] J. Li, X. Li, J. Lu, B. Cao, and J. Sun, "A novel robust hybrid control strategy for a quadrotor trajectory tracking aided with bioinspired neural dynamics," *Applied Sciences*, vol. 14, no. 20, p. 9592, Oct. 2024, doi: 10.3390/app14209592.

[18] J. Dou, Y. Wu, D. Xie, and T. Zhang, "Trajectory tracking and formation control of quadrotor UAVs based on modified super twisting sliding mode method," *Scientific Reports*, vol. 15, no. 1, p. 24039, Jul. 2025, doi: 10.1038/s41598-025-10333-2.
[19] E. A. Gedefaw, C. M. Abdissa, and L. N. Lemma, "An improved trajectory tracking control of quadcopter using a novel sliding mode control with fuzzy PID surface," *PLOS ONE*, vol. 19, no. 11, p. e0308997, Nov. 2024, doi: 10.1371/journal.pone.0308997.
[20] J.-J. Bae and J.-Y. Kang, "Quaternion-based robust sliding-mode controller for quadrotor operation under wind disturbance," *Aerospace*, vol. 12, no. 2, p. 93, Jan. 2025, doi: 10.3390/aerospace12020093.
[21] H. Ma, Y. Gao, Y. Yang, and S. Xu, "Improved nonlinear model predictive control based fast trajectory tracking for a quadrotor unmanned aerial vehicle," *Drones*, vol. 8, no. 8, p. 387, Aug. 2024, doi: 10.3390/drones8080387.
[22] J. Qi, J. Chu, Z. Xu, C. Huang, and M. Zhu, "NN-based parallel model predictive control for a quadrotor UAV," *Processes*, vol. 11, no. 6, p. 1706, Jun. 2023, doi: 10.3390/pr11061706.
[23] H. Wang and Z. Chen, "Trajectory tracking control for quadrotor slung-load system: A fuzzy pre-compensation active disturbance rejection control approach," Jan. 2026, doi: 10.1177/09596518251404227.
[24] D.-M. Lee, J.-H. Jung, Y.-S. Sim, and G.-W. Kim, "Tracking control of quadrotor micro aerial vehicles using efficient nonlinear model predictive control with C/GMRES optimization on resource-constrained microcontrollers," *Electronics*, vol. 14, no. 14, p. 2775, Jul. 2025, doi: 10.3390/electronics14142775.
[25] Z. Zhou, "Model predictive control (MPC) for quadcopter UAV dynamics: A technical overview of obstacle avoidance," *International Core Journal of Engineering*, vol. 11, no. 8, p. 13, 2025.
[26] A. Çakan, "Multi-objective grey wolf optimizer-tuned LQR attitude control of a three-DOF hover system," *Biomimetics*, vol. 11, no. 3, p. 215, Mar. 2026, doi: 10.3390/biomimetics11030215.
[27] R. Macit and S. M. Özer, "Flight control of a 3DOF quadrotor hover by state feedback with integral action," *Journal of Scientific Reports-A*, no. 063, pp. 121–138, Dec. 2025, doi: 10.59313/jsr-a.1798667.
[28] D. Dhahbane, A. A. Messaoui, A. Kourtel, and A. Ferah, "Optimised fractional order PID controller for a 3 DOF Hover quadrotor: experimental validation and robustness analysis," *International Journal of Automation and Control*, vol. 20, no. 1, pp. 1–20, 2026, doi: 10.1504/IJAAC.2026.150559.
[29] G. E. M. Abro and A. M. Abdallah, "Swarm coordination and trajectory tracking in quadrotor UAVs using fractional-order PID control strategy," *IEEE Transactions on Automation Science and Engineering*, vol. 23, pp. 2995–3008, 2026, doi: 10.1109/TASE.2025.3582119.
[30] M. Brahimi, I. Haouam, R. Bouddou, O. Almomani, A. O. Salau, and I. Hunko, "A hybrid PSO–FPA metaheuristic algorithm for ultra-low sidelobe and high-directivity synthesis of concentric circular antenna arrays for advanced radar applications," *Scientific Reports*, vol. 16, no. 1, p. 7037, Feb. 2026, doi: 10.1038/s41598-026-36315-6.
[31] S. Obukhov, E. Ahmed, D. Davydov, T. Alharbi, A. Ibrahim, and Z. Ali, "Modeling wind speed based on fractional ornstein-uhlenbeck process," *Energies*, vol. 14, no. 17, p. 5561, Sep. 2021, doi: 10.3390/en14175561.

BIOGRAPHIES OF AUTHORS

Imam Barket Ghiloubi received the master's degree from the Department of Electrical Engineering/Automatic and Industrial Informatics in the University of Biskra, Algeria, in 2020. He is a Ph.D. student in the Identification, Control, Command and Communication Laboratory LI3cub. He joined the Electrical Engineering Department and Informatics Sciences Department at the same university as an assistant professor. His research interests are focusing on the topics of artificial intelligence and autonomous systems control like ground robots, helicopters, drones, multi-agent systems and optimization techniques applied in the same topics. He can be contacted at imam.ghiloubi@univ-biskra.dz.

Latifa Abdou received the Engineering degree from the institute of electronic in the University of Batna, Algeria, in 1994 and the magister degree at the same institute in 1999. She received her Ph.D. degree in 2009. After graduation, she joined the Electrical Engineering Department at the University of Biskra, then Technological Faculty at the University Mostefa Ben Boulaïd (Batna 2), where she is a professor. She is the head of a research team in the Identification, Control, Command and communication Laboratory LI3cub. Her research interests are focusing on distributed detection systems and the application of heuristic methods in optimisation problems. In addition, she also investigates questions related to automatic systems. She can be contacted at l.abdou@univ-batna2.dz.