Optimizing Drone Control for Wind Turbine Inspection

CHABIR AMAL, ABID AICHA, BEN HMED MOUNA
Research Unit of Energy Processes Environment and Electrical Système,
National Engineering School of Gabés,
University of Gabés,
Gabés 6029,
TUNISIA

Abstract: - The paper discusses using drones for monitoring and diagnosis of issues related to wind turbine performance. The drone system helps find out why wind turbines aren't working as well as they should. It can spot problems like broken parts or weather issues. Plus, it keeps an eye on birds around the turbines, checking for nests on the structures. Using drones like this makes turbine maintenance and environmental checks much easier and cheaper, and it's also safer than the old ways. To make sure the drone works perfectly, we used MATLAB to create a model of it, using both PID and FOPID controllers to control the drone. Turns out, the FOPID controller is more accurate and does a better job of keeping the drone stable than the regular PID controller.

Key-Words: - Drone, FOPID Controller, PID Controller, Modeling, Control System, UAV Nonlinear Dynamics, Stabilization.

Received: October 18, 2024. Revised: November 25, 2024. Accepted: February 21, 2025. Published: April 7, 2025.

1 Introduction

To days, it's tough to find problems with wind turbines, especially when it comes to the blades. And to make things more complicated, birds often nest in and around the turbines. To help solve this, we've started using drones to keep an eye on the areas where the turbines are located, [1]. This makes inspections faster and more effective.

It is important to understand how drones work, before we dive in. These unmanned aerial vehicles are pretty sophisticated, combining things like aerodynamics, control systems and sensors to do all sorts of jobs, [2].

If you think about what drones are capable of, you start to get a sense of their potential and how much clever engineering goes into making them work so well. After reading a bunch of articles by the pros [3], we have concluded that a drone or UAV, for an Unmanned Aerial Vehicle is essentially a fancy robot that can fly itself. To makes them incredibly versatile for things like surveillance, package delivery, and research, you can control them remotely, or even program them to fly autonomously.

The first thing you need to do is create a model of how it works When you are trying to build a self-flying drone. That means getting to grips with the equations that describe how the drone moves and responds. How good this model is will directly affect how realistic the simulation is, and ultimately, how well your control system works.

Given the significant number of factors affecting the dynamics of a quadrotor such as aerodynamic effects, gravity, gyroscopic effects, friction, and moment of inertia it is one of the most complex flying systems.

A quadrotor is a type of VTOL (Vertical Take-Off and Landing) drone, meaning it can take off and land vertically, unlike airplanes, for example, which require a runway to gain momentum. A quadrotor consists of the following components: a cross-shaped (X) or plus-shaped (+) frame, four propellers attached to motors (most commonly brushless DC motors) positioned at the ends of the structure, a control board located at the center of the frame to process sensor data and control the motors, one or more batteries, and various sensors (such as accelerometers, ultrasonic sensors, etc.). A quadrotor is a type of VTOL (Vertical Take-Off and Landing) drone, meaning it can take off and land vertically, unlike airplanes, for example, which require a runway to gain momentum. A quadrotor consists of the following components: a cross-shaped (X) or plus-shaped (+) frame, four propellers attached to motors (most commonly brushless DC motors) positioned at the ends of the structure, a control board located at the center of the frame to process sensor data and control the motors, one or

more batteries, and various sensors (such as accelerometers, ultrasonic sensors, etc.).

2 Dynamic Model of a Quadrotor

In conventional helicopters, the main rotor generates a reactive torque as it spins, which causes the aircraft to rotate on itself. To counteract this torque, helicopters are typically equipped with a tail rotor that produces lateral thrust or a second rotor paired with the main rotor, rotating in the opposite direction.

In the case of a quadrotor, to prevent the aircraft from spinning on itself, it is essential for one pair of opposing propellers on the arms to rotate in the same direction, while the other pair rotates in the opposite direction. This configuration counteracts the reactive torque effect.

Figure 1 shows that a quadrotor is defined in space by six degrees of freedom (6 DOF): three rotations and three translations. To control these six states, it is necessary to skillfully adjust the power of the motors. The quadrotor is always leaning a bit towards the rotor that's spinning the slowest. That leaning action is actually how we control the drone's angle, allowing it to tilt and move in the direction we want it to go, [4]. Even though we only have four rotors to control the drone, we need to keep track of six different things to know exactly what it's doing – things like where it is in space and which way it's pointing. Because we have fewer controls than things we need to control, engineers call the quadrotor an 'underactuated' system, [5].

.

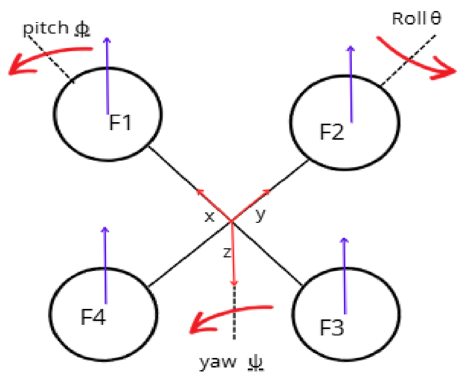

Fig. 1: The Six Degrees of Freedom of the Quadcopter

Getting a realistic computer model of a quadrotor drone in flight is a surprisingly difficult problem. The way a quadrotor moves is incredibly complex and sensitive (engineers call this 'highly nonlinear'), and all the different factors affecting it are tangled up together.

To make the problem manageable, researchers often rely on simplifying assumptions. For example [6], [7]:

- We assume the drone is perfectly balanced and that its weight is centered and aligned with our reference point.
- The drone's body is assumed to be a perfectly rigid and symmetrical shape.
- The propellers are considered perfectly rigid, allowing us to ignore any slight bending at high speeds.
- We assume the lift and drag created by the propellers increase proportionally to the square of their rotational speed.

To achieve this, we will define two reference frames: a fixed reference frame attached to the Earth Km and a moving reference frame attached to the quadrotor's center of gravity K. Instead, the moving frame is attached to the drone and tracks its motion, which makes it perfect for a more intuitive and computationally efficient expression of aerodynamic forces, motor thrust, and rotational dynamics. The fixed frame, which is frequently Earth-centered, offers an absolute reference for position and orientation, enabling global trajectory planning, navigation, and external force modeling, such as gravity and wind disturbances.

Both frames must be used to accurately describe the drone's motion. It is feasible to build control techniques in the local frame and study the drone's behavior in global coordinates by converting variables between these frames. For example, to estimate position and navigate, sensor measurements from an onboard IMU (Inertial Measurement Unit) must be transformed from the moving frame to the fixed frame. Similarly, directives for the drone's actuators in the moving frame must be converted from control inputs calculated in the fixed frame.

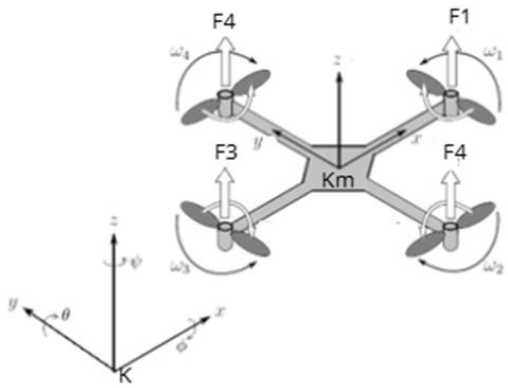

Fig. 2: Fixed and Moving Frames of the Quadcopter

The reference frames take place in modeling a quadcopter drone as it is illustrated in Figure 2. The four rotors, F1, F2, F3, and F4, are part of the drone. Each rotor generates an upward thrust Fi and rotates at an angular velocity ωi. Km is the center of gravity of the quadcopter. There are two coordinate systems: the inertial (fixed) coordinate system K (x, y, z) and the body (moving) coordinate system Km. The inertial coordinate system gives a global reference for position and orientation, something essential for navigation and control, particularly in quadcopters. The body frame shall be fixed to the drone and shall move with it, thus simplifying the analysis of forces and torques acting on the system. The rotational motions of the quadcopter will be defined by three Euler angles: roll (φ), pitch (θ), and yaw (ψ), denoting rotations about the x, y, and z axes, respectively. This dual-frame representation is fundamental to the development of flight dynamics models and control strategies aimed at stabilization and trajectory tracking.

To transition from the fixed reference frame to the moving reference frame, we define a transformation matrix TR that represents the position and orientation of the moving frame relative to the fixed frame

$$\mathrm{TR} = \begin{bmatrix} \mathrm{K} & \mathrm{E} \\ 0 & 1 \end{bmatrix} \quad (1)$$

where,

K: the rotation matrix of the moving object.

$$\mathrm{E} = [\mathrm{x} \;\; \mathrm{y} \;\; \mathrm{z}\,]^{\mathrm{T}} \quad (2)$$

is the position vector.

To determine the elements of the matrix K, we will use Euler angles. Initially, the moving reference frame coincides with the fixed reference frame, then it undergoes three rotational movements: a rotation around the x-axis with a roll angle ϕ, a rotation around the y-axis with a pitch angle θ, and finally a rotation around the z-axis with a yaw angle ψ.

We assume that:

$$-\frac{\pi}{2} < \emptyset < \frac{\pi}{2}, \;\; -\frac{\pi}{2} < \theta < \frac{\pi}{2}, \;\; -\pi < \Psi < \pi \quad (3)$$

$$\mathrm{K}=\mathrm{ROTz}(\Psi).\mathrm{ROTy}(\theta).\mathrm{ROTx}(\phi) \quad (4)$$

$$\begin{bmatrix} C\Psi & -S\Psi & 0 \\ S\Psi & C\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix}\begin{bmatrix} C\emptyset & 0 & S\emptyset \\ 0 & 1 & 0 \\ -S\emptyset & 0 & C\emptyset \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} C\Psi C\theta & S\emptyset S\theta S\Psi - S\Psi C\emptyset & C\emptyset S\theta C\Psi - S\Psi S\emptyset \\ S\Psi C\theta & S\emptyset S\theta S\Psi + C\Psi C\theta & C\emptyset S\theta S\Psi - S\emptyset C\Psi \\ -S\theta & S\emptyset C\theta & C\emptyset C\theta \end{bmatrix} \quad (6)$$

With : C:cos, S:sin.

For the angular velocities, we define the rotational velocities Ω1, Ω2, and Ω3 in the fixed frame. These velocities are expressed in terms of the velocities $\dot{\emptyset}$,$\dot{\theta}$ and $\dot{\Psi}$ in the moving frame. Initially, the two frames coincide, and a roll rotation occurs. Then, a pitch rotation follows, and the velocity vector must be expressed in the fixed frame. Therefore, we multiply the vector $\dot{\theta}$ by $\mathrm{ROT_X}(\emptyset)^{-1}$. Finally, we have the yaw rotation, and to express the velocity vector $\dot{\Psi}$ in the fixed frame, we multiply it by$[\mathrm{ROT_y}(\theta)\mathrm{ROT_x}(\phi)]^{-1}$, [6].

It is assumed that the quadrotor undergoes small roll and pitch rotations, so we can approximate as follows: cos(φ) = cos(θ) = 1 and sin(φ) = sin(θ) = 0. Therefore, the angular velocity can be written as follows:

$$\Omega=\begin{bmatrix} \dot{\emptyset} \\ \dot{\theta} \\ \dot{\Psi} \end{bmatrix} \quad (7)$$

A quadrotor's motion and stability are controlled by several physical factors. Aerodynamic drag, thrust, and gravity are the main forces affecting the quadrotor. Aerodynamic drag inhibits motion and lowers efficiency, whereas the thrust generated by the rotors fights gravity to sustain lift. Apart from these forces, the differential thrust between the rotors causes moments or torques in the quadrotor. The roll, pitch, and yaw motions produced by these moments are crucial for regulating the orientation of the quadrotor.

In the quadrotor’s dynamics, gyroscopic effects play a notable role. The rotating rotors generate gyroscopic forces that resist changes in the orientation of the quadrotor, especially during rapid rotations or movements. These effects must be carefully considered in the control system, as they can cause unintended changes in angular momentum and impact stability, especially during aggressive movements or in the presence of external disturbances such as wind. To ensure the quadrotor’s stability and performance ,you must be understanding these forces, moments, and gyroscopic effects is critical for developing accurate models and robust control strategies, [8].

One of the most popular approaches for robot modeling is the Newton-Euler method, which is explained in (8), [9].

$$\begin{cases} \dot{E} = v \\ m\ddot{E} = F_f + F_t + F_g \\ K = KA(\Omega) \\ I\Omega = M_{gm} + M_f + M_a + M_{gh} \end{cases} \quad (8)$$

where,

E: is the quadcopter's position vector. .

m: the quadrotor's total mass..
Ω : the vector of angular velocity represented in the fixed frame.
K: the matrix of rotation.
I : the matrix of symmetric inertia .

$$I = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \quad (9)$$

$A(\Omega)$: The vector Ω's antisymmetric matrix is provided by,

$$A(\Omega)=\begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix} \quad (10)$$

F_f: the total thrust force generated by the four rotors:

$$F_f=K[0 \quad 0 \quad \sum_{i=1}^{4} F_i]^T \quad (11)$$

F_t : the drag force along the axes (x,y,z) is given as follows:

$$F_t = \begin{bmatrix} -G_{ftx}\dot{x} \\ -G_{fty}\dot{y} \\ -G_{ftz}\dot{z} \end{bmatrix} \quad (12)$$

F_g: this is the force of gravity, is given by:

$$F_g = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} \quad (13)$$

M_g: These are the moments due to the rotations of the quadrotor on itself, they are described by the following relationship:

$$M_{gm} = \Omega \wedge J_S \Omega \quad (14)$$

with J_s: the inertia matrix of the system.

M_f: moment caused by thrust and drag forces

$$M_f = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} \quad (15)$$

M_a : Aerodynamic friction induces moments:

$$M_a = \begin{bmatrix} G_{fax}\,\dot{\emptyset}^2 \\ G_{fay}\,\dot{\theta}^2 \\ G_{faz}\,\dot{\Psi}^2 \end{bmatrix} \quad (16)$$

with G_{fax}, G_{fay} *and* G_{faz} the aerodynamic friction coefficients and $\emptyset, \theta$ and Ψ the speeds angular.

M_{gh}: These are the Moments due to the rotations of the propellers on their axes, they are defined as follows:

$$M_{gh}=\sum_{i=1}^{4} \Omega \wedge I_r[0 \quad 0 \quad -1^{i+1}w_i] \quad (17)$$

with I_r : Rotor inertia , $\bar{\Omega}=w_1w_2w_3w_4$ (18)
w_i : the rotation speed of the motors.

We can now pose a complete model governing a quadcopter:

$$\begin{cases} \ddot{\emptyset} = \dot{\theta}\dot{\Psi}\frac{I_y-I_z}{I_x} - \frac{J_r}{I_x}\bar{\Omega}\dot{\theta} - \frac{G_{fax}}{I_x}\dot{\emptyset}^2 + \frac{U_2}{I_x} \\ \ddot{\theta} = \dot{\emptyset}\dot{\Psi}\frac{I_z-I_x}{I_y} - \frac{J_r}{I_y}\bar{\Omega}\dot{\emptyset} - \frac{G_{fay}}{I_y}\dot{\theta}^2 + \frac{U_3}{I_y} \\ \ddot{\Psi} = \dot{\theta}\dot{\emptyset}\frac{I_x-I_y}{I_z} - \frac{J_r}{I_x}\bar{\Omega}\dot{\theta} - \frac{G_{faz}}{I_z}\dot{\Psi}^2 + \frac{U_4}{I_z} \\ \ddot{x} = \frac{cos\emptyset sin\theta cos\Psi + sin\Psi sin\emptyset}{m}U_1 - \frac{G_{ftx}}{m}\dot{x} \\ \ddot{y} = \frac{cos\emptyset sin\theta sin\Psi - sin\emptyset cos\Psi}{m}U_1 - \frac{G_{fty}}{m}\dot{y} \\ \ddot{Z} = \frac{cos\emptyset cos\theta}{m}U_1 - \frac{G_{ftz}}{m}\dot{z} - g \end{cases} \quad (19)$$

with,

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} B & B & B & B \\ 0 & -LB & 0 & LB \\ -LB & 0 & LB & 0 \\ D & -D & D & -D \end{bmatrix} \quad (20)$$

From the modeling obtained in equation (19), we concluded that the quadrotor is a nonlinear, underactuated system with strong interactions between its states. In the next party, we will implement a control strategy using optimal ordering.From the modeling obtained, we concluded that the quadrotor is a nonlinear, underactuated system with strong interactions between its states. In the next party, we will implement a control strategy using optimal ordering.

2.1 Drone Control

The goals of drone control are robustness, stability, and accurate trajectory tracking in dynamic situations. To manage disturbances, outside uncertainties, and nonlinear system dynamics, sophisticated control mechanisms must be put into place. In a variety of applications, including surveillance, industrial inspections, and autonomous navigation, efficient control guarantees the best possible drone performance. Two control methods, PID and FOPID controllers, are getting a lot of buzz in the drone world. That's because they can boost stability, make the drone react faster, and adapt to tough flying conditions, [10].

The Fractional Order Proportional-Integral-Derivative (FOPID) controller is basically a souped-up version of the regular PID controller. It uses some advanced math, called fractional calculus, to give it extra capabilities, [11]. In contrast to the traditional PID controller, which utilizes integer-order integrals and derivatives as illustrated in (21), the FOPID uses fractional-order operators, offering two extra tuning parameters: the fractional derivative order (μ) and the fractional integral order (λ).(21) and (22) express the transfer function of PID and FOPID controllers, respectively, H(s) and F(s).

$$H(s)=G_p + G_i s^{-1} + G_d s \quad (21)$$

$$F(s) = G_P + G_i s^{-\lambda} + G_d s^{\mu} \quad (22)$$

where:
G_p: Proportional gain.
G_i: Integral gain.
G_d: Derivative gain.
λ: Order of the fractional integral ($0<\lambda\leq 1$).
μ: Order of the fractional derivative ($0\leq\mu\leq 1$).

This enhancement offers greater flexibility and accuracy in system control, particularly for complex or nonlinear systems and provides the following advantages, [12]:

Proportional Control (G_p): Reduces the rise time and ensures a quick response to error. Fractional Integral Control ($G_i s^{-\lambda}$): Accumulates error over time with fractional dynamics, improving steady-state accuracy and reducing steady-state error.

Fractional Derivative Control ($G_d s^{\mu}$): Anticipates future error behavior with fractional sensitivity, enhancing system stability and transient response.

By tuning G_p, G_i , G_d , λ , and μ, the FOPID controller offers superior performance in terms of robustness, stability, and adaptability compared to the classical PID controller. It is particularly effective in controlling systems with nonlinearity, time delays, or highly dynamic behaviors.

FOPID controllers are widely used in fields such as robotics, aerospace, power systems, and process control, where precise and adaptive control strategies are essential.

3 Simulation and Results Analysis

In this section, we tested our system by applying four inputs, namely x, y, z and ψ, to verify that our system stabilizes and converges towards the given trajectory as illustrated in Table 1. We applied two control strategies, the classical PID controller and the FOPID controller, to determine the most effective one.

Table 1. The values of the FOPID and PID command

	X	y	z	ϕ	θ	Ψ
PID	Kp=100	100	98	10	10	1
	Ki=21.6	23	27.6	0.4	0.7	0.4
	Kd=20	24	27	0.56	1.8	0.65
FOPID	λ= 0.5	0.6	0.98	0.8	0.65	0.98
	μ= 0.4	0.4	0.87	0.75	0.7	0.87

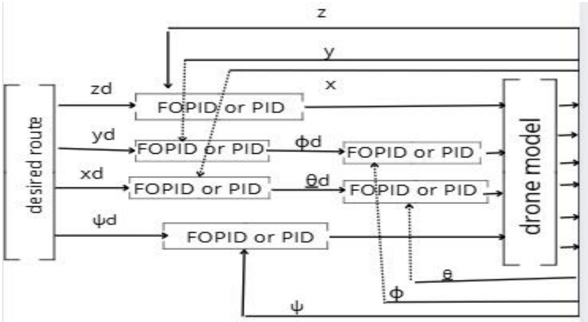

Fig. 3: FOPID and PID Control Scheme for a Drone

The control architecture of a drone employing a Fractional-Order PID (FOPID) or Proportional-Integral-Derivative (PID) controller is shown in Figure 3. Reference signals (xd,yd,zd,ψd), which indicate the drone's desired location and yaw angle, are used by the system to track a predetermined path. A PID or FOPID controller processes each reference signal to produce the proper control inputs. To guarantee precise trajectory tracking, the controllers control the orientation angles (ϕ,θ,ψ) and position (x, y, z).

The control outputs are fed into the drone model, which represents the dynamics of the UAV (Unmanned Aerial Vehicle). The feedback loop ensures continuous correction by comparing the actual state variables with the desired values, minimizing errors, and enhancing stability. The use of FOPID controllers offers improved robustness and flexibility compared to classical PID controllers, particularly in handling dynamic uncertainties and external disturbances.

Table 2 presents the values used during the simulation of the system dynamics in Matlab-Simulink.

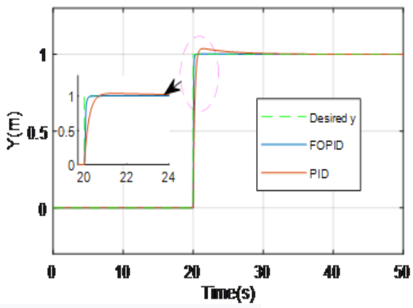

Fig. 4: Response in position x

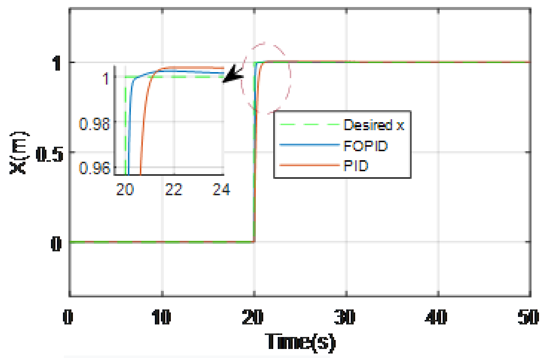

Fig. 5: Response in position y

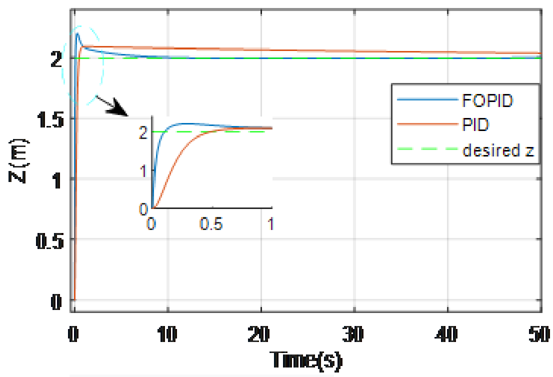

Fig. 6: Response in position z

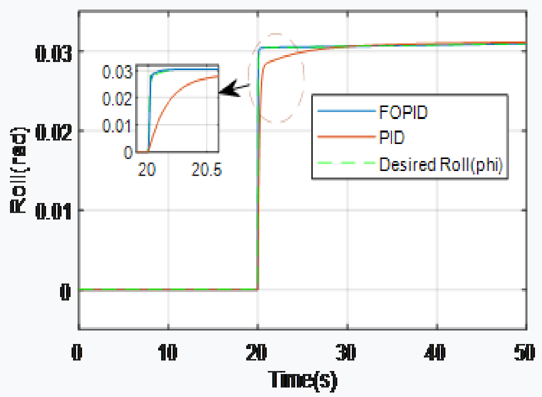

Fig. 7: Angled response Roll

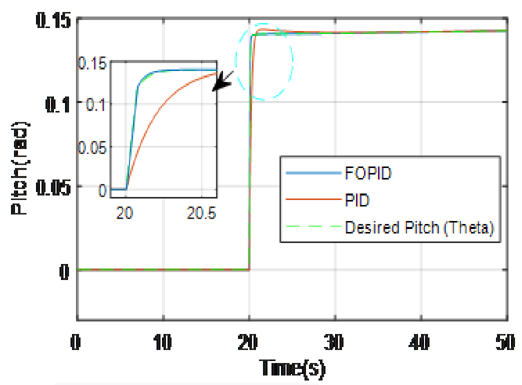

Fig. 8: Angled response Pitch

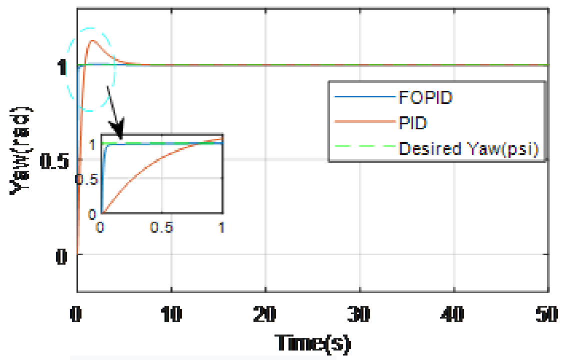

Fig. 9: Angled response Yaw

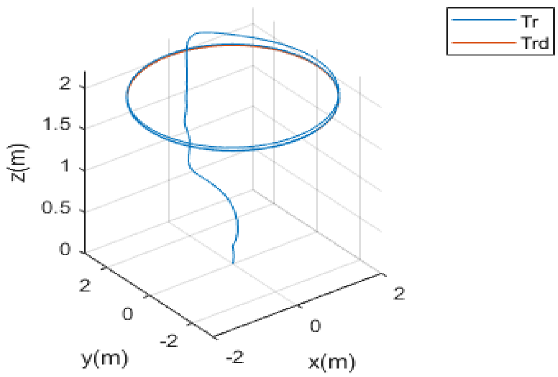

Fig. 10: control the route of the drone with the defined route

Figure 4, Figure 5 and Figure 6 compare the performance of a Proportional-Integral-Derivative (PID) controller and a Fractional Order PID (FOPID) controller in tracking the desired trajectory, denoted as "Desired x." The x-axis represents Time (s), while the y-axis indicates the system output position, X (m), Y(m), and Z(m).

The green dashed line illustrates the reference or desired output position, showing a sudden increase at approximately t=20 s , indicating a step change in the system's set point.

The response of the conventional PID controller is represented by the orange solid line. It shows a noticeable overshoot and a slower settling time, taking longer to stabilize around the desired position

The response of the FOPID controller is shown with the blue solid line. Compared to the standard PID, the FOPID exhibits reduced overshoot and a faster settling time, reaching the steady-state value closer to the desired trajectory.

The inset in the figure provides a magnified view of the transient response between t=20 s, and t=25 s. This detailed section highlights the performance difference between the two controllers.The PID controller overshoots before converging, whereas the FOPID controller tracks the intended trajectory with little deviation.

In terms of transient response, the figure shows that the FOPID controller performs better than the conventional PID controller, providing increased accuracy, less overshoot and quicker convergence to the target position.

Figure 7, Figure 8 and Figure 9 show how the Roll, Pitch, and Yaw angles respond (in radians) over time (in seconds) for a control system that compares two controllers: Proportional-Integral-Derivative (PID) and Fractional Order Proportional-Integral-Derivative (FOPID). A dashed green line indicates the desired yaw angle (ψ). The results show that the FOPID controller achieves a faster settling time and a reduced overshoot compared to the PID controller, as seen from the main plot. The transient response during the first one-second period is highlighted in the zoomed-in section. It shows that the PID controller shows a noticeable overshoot and slower convergence to the desired value, whereas the FOPID controller demonstrates a smoother and faster response, closely following the desired yaw angle. These findings underscore the superior performance of the FOPID controller in terms of precision and stability when compared to the conventional PID controller.

Figure 10 illustrates a three-dimensional trajectory comparison for a drone's path, displaying two curves:

Tr (blue): Represents the actual trajectory followed by the drone.

Trd (red): Depicts the desired or reference trajectory.

The actual trajectory (Tr) indicates a complex movement, starting with a spiral ascent or descent before aligning with the circular reference trajectory at a specific altitude.

The desired trajectory (Trd) is a smooth, well-defined circular path in the horizontal plane, representing the intended navigation route.

The initial deviations between Tr and Trd suggest potential errors, disturbances, or delays in the drone's ability to follow the desired trajectory during the early phase.

In the final phase, the actual trajectory (Tr) converges toward the desired trajectory (Trd), demonstrating improved control performance and stabilization over time.

3.1 Stability Analysis

The simulation results prove that the PID controller dynamically adjusts the motor rotation speed in real-time to correct deviations in orientation or position. For instance, if the drone tilts forward, the controller will increase the speed of the rear motors to bring it back to a horizontal position. The proportional, integral, and derivative actions work together to ensure a rapid response, eliminate residual errors, and anticipate variations, thereby ensuring effective drone stabilization, [13].

In summary, the PID as well as the FOPID controller is crucial for drone stabilization, as it dynamically adjusts motor commands based on measured errors, thereby ensuring stable and controlled flight as shown in Figure 4, Figure 5, Figure 6, Figure 7 and Figure 8 that have validated that the system is stable, [14].

This analysis highlights the drone's ability to correct deviations and adhere to the intended trajectory, which is critical for evaluating the effectiveness of the guidance, navigation, and control systems.

Table 2. Values of the system

Parameter	Value
m	1 kg
g	9.8 m/s^2
L	0.25 m
B	2984.10^{-5}N/rad/s
D	$3.2320x10^{-7}$N/rad/s
Jr	$2.8385x10^{-5}kgm^2$
Ix	$3.8278x10^{-3}kgm^2$
Iy	$3.8278x10^{-3}kgm^2$
Iz	$7.6566x10^{-3}kgm^2$
Gfax	5.5670x10-4N/rad/s
Gfay	5.5670x10-4N/rad/s
Gfaz	$6.3540x10^{-4}$N/rad/s
Gftx	$5.5670x10^{-4}$N/m/s
Gfty	$5.5670x10^{-4}$N/m/s
Gftz	$6.3540x10^{-4}$N/m/s

4 Conclusion

The study found that the performance of Fractional Order Proportional-Derivative (FOPID) and Proportional-Integral-Derivative (PID) controllers in drone systems are widely used for stability and trajectory control. The results highlight the following:

The PID Controllers are easy to implement and tune, offering adequate performance in standard conditions. However, in dynamic or nonlinear environments, they may face difficulties.

The FOPID Controllers are Provide enhanced robustness and adaptability due to fractional calculus, leading to better handling of disturbances and uncertainties. But, they are more complex to design and compute.

FOPID controllers perform better in situations that call for accuracy and stability, particularly when there are outside disturbances present. Building on the results, using these controllers to improve environmental monitoring systems is a promising direction for further study.

In particular, wind turbines can be monitored in challenging environmental conditions using drones fitted with FOPID controllers.

Declaration of Generative AI and AI-assisted Technologies in the Writing Process

During the preparation of this work the authors used Grammarly and Quillbot in order to correct the language and write texts. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References:

[1] Llabani, A. and O. Lubonja, Integrating UAV Photogrammetry and Terrestrial Laser Scanning for the 3D surveying of the Fortress of Bashtova. *WSEAS Transactions on Environment and Development,* 2024. 20: p. 306-315. https://doi.org/10.37394/232015.2024.20.30.

[2] Alshbatat, A.I.N., M. Awawdeh, and U.A. EMIRATES, Drone-Based Adaptive Hybrid Techniques for Improving Face Detection and Recognition. *WSEAS Transactions on Computer Research,* 2024. 12: p. 516-523. https://doi.org/10.37394/232018.2024.12.50.

[3] Park, O. and H.-S. Shin., Dynamics Modeling of Multirotor type UAV with the Blade Element Momentum Theory and Nonlinear Controller Design for a Wind Environment, in *AIAA SCITECH 2024 Forum.* 2024. doi: doi.org/10.2514/6.2024-2875.

[4] Hien, N. and P. Diem, A Model-Based Design to Implement Controllers for Quadrotor Unmanned Aerial Vehicles. *WSEAS Transactions on Systems,* vol.18, pp. 45-61, 2019.

[5] C. Rus, E. Lupulescu, M. Leba and M. Risteiu, "Advanced Mathematical Modeling and Control Strategies for Autonomous Drone Systems," 2024 *25th International Carpathian Control Conference (ICCC)*, Krynica Zdrój, Poland, 2024, pp. 1-6. doi: 10.1109/ICCC62069.2024.10569613.

[6] Schmidt, D.K., W. Zhao, and R.K. Kapania. Flight-dynamics and flutter modeling and analyses of a flexible flying-wing drone-invited. in *AIAA Atmospheric Flight Mechanics Conference. 2016,* 4-8 January 2016San Diego, California, USA doi.org/10.2514/6.2016-1748.

[7] Zabidin, Y.A.A., M.F. Pairan, and S.S. Shamsudin, Dynamic modelling and control for quadcopter uav with labview and x-plane flight simulator. *Journal of Complex Flow,* 2020. 2(2): pp. 19-26, [Online]. https://jcf.fazpublishing.com/index.php/jcf/article/view/24 (Accessed Date November 13, 2024).

[8] Zhang, Y., et al., A simplified FE modeling strategy for the drop process simulation analysis of light and small drone. *Aerospace,* 2021. 8(12): p. 387, doi.org/10.3390/aerospace8120387.

[9] L. Shan, R. Miura, T. Kagawa, F. Ono, H. -B. Li and F. Kojima, "Machine Learning-Based Field Data Analysis and Modeling for Drone Communications," in *IEEE Access,* vol. 7, pp. 79127-79135, 2019. doi: 10.1109/ACCESS.2019.2922544.

[10] Yauri, R., S. Fernandez, and A. Aquino, Control of Autonomous Aerial Vehicles to Transport a Medical Supplies. *WSEAS Transactions on Systems,* 2024. 23: p. 73-81. https://doi.org/10.37394/23202.2024.23.8.

[11] R. Betala and S. P. Nangrani, "Comparison of Performance of Fractional Order PID Controller with Conventional Controller for Industrial Applications," *2023 IEEE International Conference on Integrated Circuits and Communication Systems (ICICACS),* Raichur, India, 2023, pp.16. doi: 10.1109/ICICACS57338.2023.10099955.

[12] Espinoza, J., Hakim, N., Tan, D., Wilson, T., Bingi, K., Khan, E., & Masrura, S. (2023, December). Fractional-order pid control of quadrotor drone. In 2023 Innovations in Power and Advanced Computing Technologies (i-PACT) (pp. 1-6). IEEE. doi: 10.1109/iPACT58649.2023.10434503.

[13] Skraparlis, A. N., Ntalianis, K. S., Ntaliani, M. S., Ntalianis, F. S., & Mastorakis, N. E. (2024). Detecting Indoor Tiny Autonomous Malicious Drones within Critical Infrastructures: An Innovative Algorithm based on Harmonic Radar-Equipped Mini-Drones. *WSEAS Transactions on Information Science and Applications,* 21, 466-479, https://doi.org/10.37394/23209.2024.21.42.

[14] Arfa, W., Jabeur, C. B., Faleh, Y., & Seddik, H. (2024). Hybrid multi-control for better drone stability. *International Journal of Modelling, Identification and Control,* 45(2-3),164-177. doi: doi.org/10.1504/IJMIC.2024.142263.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

- Chabir Amal contribute by the creation of the drone model.
- Abid Aicha has implemented the control scheme.
- Ben hamed Mouna was responsible for the writing task.

The authors equally contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

The author gratefully acknowledges the editor and referees for their valuable comments and suggestions, which greatly improve this paper. This work was supported by grants from National Engineering School of Gabes, Tunisia

Processes, Energy, Environments and Electrical Systems (PEESE) laboratory LR18ES34 at National Engineering School of Gabes, Tunisia.

Conflict of Interest

The authors have no conflicts of interest to declare.