AKADÉMIAI KIADÓ

UNIVERSITY of DEBRECEN

International Review of Applied Sciences and Engineering

13 (2022) 2, 133–147

DOI:
10.1556/1848.2021.00325

ORIGINAL RESEARCH PAPER

Neural networks on-line optimized PID controller with wind gust rejection for a quad-rotor

Chiraz Ben Jabeur[1]* and Hassene Seddik[2]

[1] University of Tunis, RIFTSI, ENSIT, University of Tunis El Manar, ISI, 2, Rue Abou Raihane Bayrouni, 2080, Ariana, Tunisia

[2] University of Tunis, Department of Electrical Engineering, RIFTSI, ENSIT, 5, Av. TahaHussein, 1008, Tunis, Tunisia

Received: June 11, 2021 • Accepted: August 27, 2021
Published online: November 11, 2021

ABSTRACT

In this paper a complete methodology of modeling and control of quad-rotor aircraft is exposed. In fact, a PD on-line optimized Neural Networks Approach (PD-NN) is developed and applied to control the attitude of a quad-rotor that is evolving in hostile environment with wind gust disturbances and should maintain its position despite of these troubles. Whereas PD classical controllers are dedicated for the positions, altitude and speed control. The main objective of this work is to develop a smart Self-Tuning PD controller for attitude angles control, based on neural networks capable of controlling the quad-rotor for an optimized performance thus following a desired trajectory. Many problems could arise if the quad-rotor is evolving in hostile environments presenting irregular troubles such as wind gusts modeled and applied to the overall system. The quad-rotor has to rapidly achieve tasks while guaranteeing stability and precision and must behave quickly with regards to decision making fronting turbulences. This technique offers some advantages over conventional control methods such as PD controllers. Simulation results are achieved with the use of Matlab/Simulink environment and are established on a comparative study between PD and PD-NN controllers founded on wind disturbances application. These obstacles are applied with numerous degrees of strength to test the quad-rotor comportment. Experimental results are reached with the use of the V-REP environment with which some trajectories are tracked and then applied on a BLADE Inductrix FPV+. These simulations and experimental results are acceptable and have confirmed the efficiency of the proposed PD-NN approach. In fact, this controller has fairly smaller errors than the PD controller and has an improved ability to reject troubles. Moreover, it has confirmed to be extremely vigorous and efficient fronting disturbances in the form of wind disturbances.

KEYWORDS

quad-rotor control, BLADE Inductrix FPV+, robustness and stability, hostile environment, trajectory tracking, disturbances rejection

1. INTRODUCTION

Up to now, flying robots enjoy great popularity and the control of these systems is the main subject in robotics research, in military and space studies. Unmanned Air Vehicle can be remotely controlled or can fly independently [1]. In fact, autonomous Unmanned Air Vehicles are becoming more and more popular. By regulating the motors power, the quad-rotor can be capable of achieving several tasks. That is to say the control of a quad-rotor is not a minor task because of the six degrees of freedom, the high nonlinearities presented in the responses, the strong coupling multivariable and the under actuated conditions, especially with only four motors. Some studies worked on the control of such a system in a wide field of applications such as trajectory tracking control and obstacle avoidance control [2, 3]. In the

*Corresponding author.
E-mail: chirazbenjabeur@gmail.com

AKJournals

mission of trajectory tracking, certain applications are based on conventional techniques such PD and PID [4–7], and others are based on artificial intelligence, such as Fuzzy Logic, Neural Networks and Neuro-Fuzzy systems [8–10]. In [8], four cooperative PD controllers were replaced by four Neural Networks with which each Neural Network imitated the actions of a PD controller and Fuzzy logic was used to adjust PID controllers' gains. In addition, in [9], fuzzy logic is used to design a robust Self-Tuning PID controller. It was a task to optimize PID gains with fuzzy logic for heading and position trajectory tracking control to manipulate the external disturbances caused by the payload weight variation for the period of flight. Moreover, in [10], the control of an unmanned aerial vehicle in tracking a moving object is exposed with the use of three fuzzy logic units that are implemented to permit the engine following the desired position and inclination.

PID controller has become an essential technical tool and is successfully applied in robotic systems and especially in quad-rotors control strategies. Nevertheless, to get ideal control effect, it is required to optimize its three parameters: proportional coefficient K_P, integral coefficient K_I, and derivative coefficient K_D before placing it into simulation.

It is relevant to affirm that many research papers deal with PD and PID parameters optimisation with fuzzy logic such as in [11–13], but not a lot use neural networks approach.

The aim of parameters optimization is to achieve the best control effect by making the controller and the controlled systems characteristics well-coordinated. If the selected parameters of the PD controller are inopportune, its control effect will be very modest. However, there are all kinds of uncertainties and nonlinearities in the quad-rotor control system, so it is hard to establish the precise traditional math model. Moreover, the traditional parameter optimization methods of the PD controller cannot promise normal work and it is difficult to realize perfect control effect. For this reason, recently, neural networks have demonstrated their growth in PD parameters optimization. In fact, the neural network can learn by itself and simulate the system parameters without knowing about the structure of the system so as to get the system rule. Currently, PID controller parameters optimized by neural network are becoming a promising topic [14–16]. In fact in [14], a robust PID controller based artificial neural network is presented for quad-rotor control. The proposed PD tuning algorithm adjusts continuously the PD parameters with tracking error minimization.

The objective of this work is using artificial intelligence permitting to conceptualize an intelligent tracking control to a quad-rotor that is exposed to wind disturbances. In this way, neural networks are used to adjust the parameters of the PD controller for the attitude control in an on line manner. This attitude control concerns the roll pitch and yaw angles. Thus, there are three neural networks for each angle, which structures are so special and contain at least one hidden layer of two neurons that are the Proportional and the Derivative functions.

Basically, the system may meet some wind disturbances in its fly, that depend on its environment, and that can affect its trajectory following. In this case, the quad-rotor must compensate these disturbances and continue its navigation without changing its trajectory.

Accordingly, the principal objective is the design and implementation of PD classical controllers and PD-NN controllers for controlling the attitude angles, based on neural networks in order to best allow the robot following a desired behavior while tracking a desired trajectory despite of wind disturbances. The quad-rotor must be robust to sudden environment change and must react quickly.

This paper is composed of seven sections. The first section contains introduction. The second one deals with dynamics modeling. The third section accounts for the model of wind gust. In the forth the PD classical control strategy and the PD-NN control strategy optimized by neural networks are detailed. The fifth section deals with both simulations and experimental results. The sixth section is dedicated to analysis. The seventh section presents conclusion.

2. DYNAMIC MODEL OF THE QUAD-ROTOR

Modeling of a quad-rotor is based on the overall model which is built with the dynamic models of the quad-rotor in addition to the DC motor dynamics that must be taken into account. A basic model of an unmanned quad-rotor [17–20] is shown in Fig. 1.

It is assumed that the front and the rear motors rotate counter clockwise when the others rotate clockwise. The sum of the thrusts of each motors is the throttle input. If the speed of the rear and the front motors is increased (decreased), the movement of the pitch is attained. If the lateral motors are used, the roll movement is achieved similarly. If the speed of the front and rear motors is increased (decreased) and the speed of the lateral motors is decreased (increased), the yaw movement is activated [18].

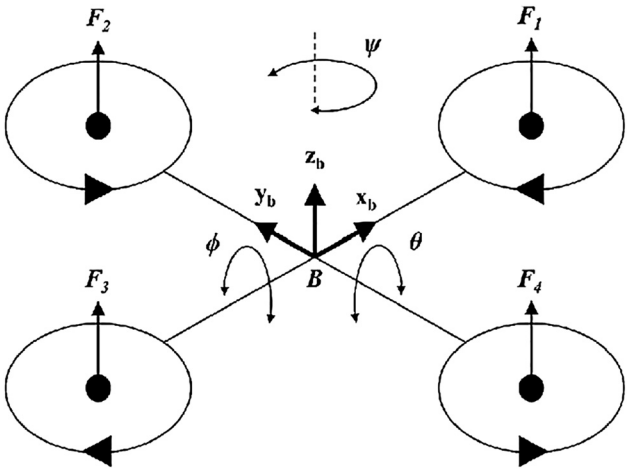

Fig. 1. Model of quad-rotor

2.1. Newton-Euler model

In this section, the Newton-Euler formalism is used to build the specific model information of the quad-rotor architecture based on the rigid-body [21].

Two frames have to be considered such as in Fig. 1

The inertial frame of the earth (E-frame)

The vehicle body-fixed frame (B-frame)

These frames are linked over three successive rotations:

Rotation around the x-axis: Roll;

Rotation around the y-axis: Pitch;

Rotation around the z-axis: Yaw.

The following suppositions have been made in this approach:

The starting point of the body-fixed frame corresponds to the vehicle body center of mass.

The body principal axes of inertia matches the axes of the B-frame.

In the body-fixed frame, the motions equations are suitably formulated based on the following reasons [21]:

The time-invariant inertia matrix.

To simplify the equations, the benefit of body symmetry is considered.

Measurements taken on-board are easily converted to body-fixed frame.

In body-fixed frame, the control forces are always given.

The generalized coordinates of the quad-rotor are:

$q = (x,\ y,\ z,\ \phi,\ \theta,\ \psi)$ where $(x,\ y,\ z)$ represents the quad-rotor center of mass relative position with regards to an inertial frame E and (ϕ, θ, ψ) are the three angles of Euler expressing the quad-rotor orientation namely the roll, the pitch and the yaw of the vehicle.

Suppose that the transitional and rotational coordinates are in the form: $\xi = (x,\ y,\ z)) \in \Re^3$ and $\eta = (\phi,\ \theta,\ \psi) \in \Re^3$.

The total transitional kinetic energy of the quad-rotor is:

$$T_{trans} = \frac{1}{2} m \dot{\xi}^T \dot{\xi} \tag{1}$$

where m is the mass of the quad-rotor.

The rotational kinetic energy is described as:

$$T_{rot} = \frac{1}{2} J \dot{\eta}^T \dot{\eta} = \frac{1}{2} \omega^T I \omega \tag{2}$$

where matrix $J = J(\eta)$ is the auxiliary matrix, I is the body inertia matrix and $\omega$is the angular speed.

The potential energy in the system is described as:

$$U = m.g.z \tag{3}$$

where z is the quad-rotor altitude and g is the acceleration due to gravity.

The Lagrangian is defined from (1)–(3) as:

$$L = T_{trans} + T_{rot} - U \tag{4}$$

Then the full quad-rotor dynamics is obtained as a function of the external generalized forces $F = (F_\xi, \tau)$ from:

$$F = \frac{d}{dt}\frac{\partial L}{\partial q} - \frac{\partial L}{\partial q} = \begin{pmatrix} m\ddot{x} \\ m\ddot{y} \\ m\ddot{z} \\ I_x\ddot{\phi} + I_x\dot{\phi} \\ I_y\ddot{\theta} + I_y\dot{\theta} \\ I_z\ddot{\psi} + I_z\dot{\psi} \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ -mg \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{5}$$

F_ξ is the translational force applied to the quad-rotor due to the throttle control input, τ represents the pitch, roll, and yaw moments.

The principal control inputs are defined as follows:

$$F_r = (0\ \ 0\ \ u)^T \tag{6}$$

where the main thrust is:

$$u = f_1 + f_2 + f_3 + f_4 \tag{7}$$

where $f_i = b.\omega_i^2$, b is a positive constant and ω_i are the angular speed of the motor i.

Then F_ξ can be written as:

$$F_\xi = R.F_r \tag{8}$$

where R is the transformation matrix representing the orientation of the quad-rotor as:

$$R = \begin{bmatrix} c\theta c\psi & s\theta s\psi & -s\theta \\ c\psi s\theta s\varphi - s\psi s\varphi & s\psi s\theta s\varphi + c\psi c\varphi & c\theta s\varphi \\ c\psi s\theta c\varphi + s\psi s\varphi & s\psi s\theta c\varphi - c\psi s\varphi & c\theta c\varphi \end{bmatrix} \tag{9}$$

With: $c = \cos, \quad s = \sin$

The generalized torques for the η variables are:

$$\tau = \left(\tau_\varphi\ \tau_\theta\ \tau_\psi\right)^T \tag{10}$$

where:

$$\tau_\varphi = l\left(f_3 - f_1\right) = bl\left(-\omega_1^2 + \omega_3^2\right) \tag{11}$$

$$\tau_\theta = l\left(f_2 - f_4\right) = bl\left(-\omega_2^2 + \omega_4^2\right) \tag{12}$$

$$\tau_\psi = \sum_{i=1}^{,} \tau_{Mi} = d\left(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2\right) \tag{13}$$

Thus the control distribution from the four actuator motors of the quad-rotor is given by:

$$\begin{bmatrix} u \\ \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -bl & 0 & bl \\ -bl & 0 & bl & 0 \\ -d & d & -d & d \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{14}$$

where l is the distance from the motors to the centre of gravity, τ_{Mi} is the torque produced by motor M_i and d and b are constants known as force-to-moment scaling factor.

The final dynamic model of the quad-rotor looks as:

$$m\ddot{\xi} + \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} = F_\xi \tag{15}$$

$$J(\eta)\ddot{\eta} + C\left(\eta,\ \dot{\eta}\right) = \tau \tag{16}$$

where,

$$F_\xi = \begin{pmatrix} c\psi s\theta c\varphi + s\psi s\varphi \\ s\psi s\theta c\varphi - c\psi s\varphi \\ c\theta c\varphi \end{pmatrix} u \tag{17}$$

And auxiliary Matrix $(J(\eta)) = J = T_\eta^T I T_\eta$

$$\text{With}: \; T_\eta = \begin{pmatrix} -s\theta & 0 & 1 \\ c\theta s\psi & c\psi & 0 \\ c\theta c\psi & -s\psi & 0 \end{pmatrix} \tag{18}$$

Thus the system has the form of an under-actuated system with six outputs $(x, y, z, \varphi, \theta, \psi)$ and four inputs $(u, \tau_\varphi, \tau_\theta, \tau_\psi)$, and the dynamic model of the quad-rotor is written as:

$$\begin{cases} \ddot{x} = (c\psi s\theta c\varphi + s\psi s\varphi)\dfrac{u}{m} \\ \ddot{y} = (s\psi s\theta c\varphi - C\psi s\varphi)\dfrac{u}{m} \\ \ddot{z} = -g + (c\theta c\varphi)\dfrac{u}{m} \\ \ddot{\varphi} = \dot{\theta}\dot{\psi}\left(\dfrac{I_y - I_z}{I_x}\right) - \dfrac{J_p}{I_x}\dot{\theta}\Omega + \dfrac{\tau_\varphi}{I_x} \\ \ddot{\theta} = \dot{\psi}\dot{\varphi}\left(\dfrac{I_z - I_x}{I_y}\right) + \dfrac{J_p}{I_y}\dot{\varphi}\Omega + \dfrac{\tau_\theta}{I_y} \\ \ddot{\psi} = \dot{\theta}\dot{\varphi}\left(\dfrac{I_x - I_y}{I_z}\right) + \dfrac{\tau_\psi}{I_z} \end{cases} \tag{19}$$

I_x, I_y, I_z are body inertia, J_p is propeller/rotor inertia, and $\Omega = (\omega_1, \omega_2, \omega_3, \omega_4)$

2.2. Actuators dynamic modeling: DC-motor

In general cases, the driving system of quad-rotors is based on an armature-controlled DC motors which are considered as servo actuators. These actuators train the system and drive it by providing torque control inputs. The architecture of the DC motor is given in Fig. 2.

Kirchhoff's voltage law is applied giving:

$$\begin{aligned} v(t) &= V_R(t) + V_L(t) + e(t) \\ &= Ri(t) + L\frac{di(t)}{dt} + K_E\omega_M(t) \end{aligned} \tag{20}$$

where $V_R(t)$ is the voltage through the resistor R and $V_L(t)$ is the voltage through the inductor L, $i(t)$ is the current of the motor, K_E is the electrical constant of the motor and ω_M is the motor angular speed.

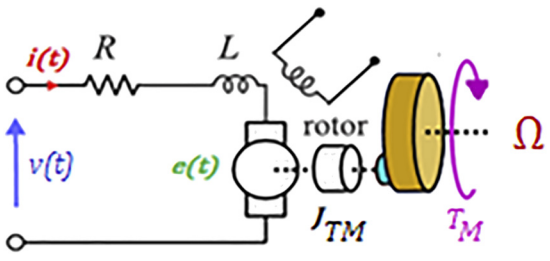

Fig. 2. DC motor architecture

Thanks to construction optimization, most of the motors used in robotics show small inductance. For this reason the inductor effect is neglected. Therefore the above equation is simplified according to this one:

$$v(t) = Ri(t) + K_E\omega_M(t) \tag{21}$$

The dynamics of the motor is described by:

$$J_{TM}\dot{\omega}_M = T_M - T_L \tag{22}$$

where J_{TM} is the total motor moment of inertia, $\dot{\omega}_M$ is the motor angular acceleration, T_M is the motor torque and T_L is the load torque.

The motor torque T_M is proportional to the electrical current i, consequently we find:

$$J_{TM}\dot{\omega}_M = -\frac{K_E K_M}{R}\omega_M - T_L + \frac{K_M}{R}v \tag{23}$$

The two constants K_E and K_M have the same value even if the units of measurement differ.

The real motor system is composed of three units which are the motor itself, the propeller and the gear box.

To take the relations between these elements into consideration, we have:

$$\omega_M T_{MP}\eta = \omega_P T_{MP} \tag{24}$$

where: T_{MP} is the motor torque in the propeller axis.

η is the conversion efficiency of the gear box which attaches the mechanical power of the motor axis P_M to the propeller axis one P_P

$$J_M\dot{\omega}_M = T_M - T_{PM} \tag{25}$$

$$J_P\dot{\omega}_P = T_{MP} - T_P \tag{26}$$

where: ω_P is the propeller angular speed and $\dot{\omega}_P$ is the propeller angular acceleration.

J_M is the rotor moment of inertia around the motor axis, J_P is the rotor moment of inertia around the propeller axis, T_M is the propeller torque in the motor axis and T_P is the propeller torque.

To calculate the dynamics of the gear box system, this equation is derived:

$$\left(J_M + \frac{J_P}{\eta N^2}\right)\dot{\omega}_M = T_M - \frac{T_P}{\eta N} \tag{27}$$

where N is the reduction ratio of the gear box and is equal to the motor speed ω_M divided by the propeller speed ω_P.

According to Eq. (23), the motor system can be modeled with the following differential equation.

$$(J_p + \eta N^2 J_M)\dot{\omega}_P = -\frac{K_E K_M}{R}\eta N^2\omega_P - d\omega_P^2 + \frac{K_M}{R}\eta N v \tag{28}$$

where d is the aerodynamic drag factor

Where the $\dot{\omega}_P$ coefficient is the total rotational moment of inertia around the propeller axis J_{TP} with:

$$J_{TP} = J_P + \eta N^2 J_M \tag{29}$$

Since the motor differential equation is non-linear, the first order Taylor series method has been adopted to linearize it around its working point as:

$$\dot{\omega}_P = A_P\omega_P + B_P v + C_P \tag{30}$$

A_P is the linearized propeller's speed coefficient, B_P is the linearized input voltage coefficient and C_P is the linearized constant coefficient.

The matrix form of the differential equation is given by:

$$\dot{\Omega}_P = A_P\Omega_P + B_P v + C_P \tag{31}$$

where $\dot{\Omega}_P$ is the propellers acceleration vector, Ω_P is the propellers speed vector, and v is the inputs voltage vector.

3. WIND GUST MODELING

If a quad-rotor undergoes a crosswind, it may be pushed far downwind or knocked over. Basically, this leads to forces f_{v_i}, acting on every rotor. These forces are expected in the air flow generated by the wind gust [22–26]. This means that the degree of these forces is a function of the additional lateral air flow coming from the wind gust and acting on the propeller as shown in Fig. 3.

Where:

$f_{mi} = b\omega_i^2$: represents the thrust of the propellers i = 1,. . ., 4.

f_{vi}: designates the additional thrust of the side wind.

α: symbolizes the angle between the axis of the propeller and the gust of side wind.

V_v: is the speed of the side wind gust.

V_h: is the speed of the wind produced by the propeller

$\widehat{V}$: is the total wind speed.

Therefore, downwind, the total thrust is given by the following expression:

$$f_{Ti} = f_{mi} + f_{vi} = 2.\rho.A.\widehat{V}.V_h \tag{32}$$

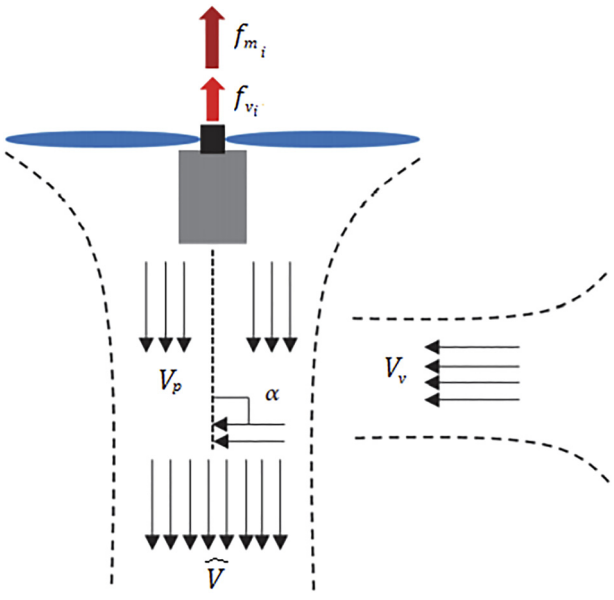

Fig. 3. Aerodynamic analysis of rotor subjected to wind disturbance

ρ: is the air density.

A: is the area of the propeller.

The relation describing the total wind speed is given by:

$$\widehat{V} = \left[(V_v\cos\alpha + V_h)^2 + (V_v\sin\alpha)^2\right]^{1/2} \tag{33}$$

When the wind comes from the right of the x-axis, then the angle $\alpha = 90°$, and this relation becomes:

$$\widehat{V} = \left[(V_h)^2 + (V_v)^2\right]^{1/2} \tag{34}$$

The wind speed V_h induced by the propellers is defined as in [24, 26] by:

$$\widehat{V} = \left[(V_v\cos\alpha + V_h)^2 + (V_v\sin\alpha)^2\right]^{1/2} \tag{35}$$

The thrust produced by the wind will therefore be expressed by:

$$f_{vi} = 2.\rho.A.\left(\frac{d.\omega_i^2}{2.\rho.A} + V_v^2\right)^{1/2} - f_{mi} \tag{36}$$

In addition, an aerodynamic drag τ_{drag} opposes the engine torque. This aerodynamic drag is given by:

$$\tau_{drag} = \frac{\rho.A}{2}V_v^2 \tag{37}$$

In the case of the quad-rotor is troubled by wind gusts, the induced forces f_{v_i} in each rotor produce additional aerodynamic wind torques acting on the propellers. These torques could be expressed by:

$$\begin{bmatrix}\tau_{v\varphi}\\ \tau_{v\theta}\\ \tau_{v\psi}\end{bmatrix} = \begin{bmatrix}(f_{v4} - f_{v2})l\\ (f_{v3} - f_{v1})l\\ \sum_{i=1}^{4}\tau_{traîné_i}\end{bmatrix} \tag{38}$$

where l is the distance between the center of the quad-rotor and the center of the propellers.

Basically, the applied gravitational force to the machine is given by:

$$f_g = -m.g.\widehat{k} \tag{39}$$

where $\widehat{k}$ refers to the unit vector coinciding with the z axis and g denotes the acceleration of gravity.

Consequently, we have:

$$\begin{cases} F = R\left(\sum_{i=1}^{4} f_{Mi} + \sum_{i=1}^{4} f_{vi}\right) + f_g \\ \tau = \tau_A + \tau_{vA} \end{cases} \tag{40}$$

where:

$\tau_A = [\tau_\varphi,\ \tau_\theta,\ \tau_\psi]^T$ and $\tau_{vA} = [\tau_{v\varphi},\ \tau_{v\theta},\ \tau_{v\psi}]^T$

From Eqs (15) and (16), we have:

$$\begin{cases} m\xi = R\left(\sum_{i=1}^{4} f_{Mi} + \sum_{i=1}^{4} f_{vi}\right) + f_g \\ J\dot{\Omega} = -\Omega\wedge J\Omega + \tau_A + \tau_{vA} - \tau_{fa} - \tau_{gh} \end{cases} \tag{41}$$

This leads to:

$$\begin{cases} \ddot{x} = -\dfrac{K_{tx}}{m}\dot{x} + (c\psi s\theta c\varphi + s\psi s\varphi)\dfrac{U_1}{m} + W_1 \\ \ddot{y} = -\dfrac{K_{ty}}{m}\dot{y} + (s\psi s\theta c\varphi - c\psi s\varphi)\dfrac{U_1}{m} + W_1 \\ \ddot{z} = -\dfrac{K_{tz}}{m}\dot{z} - g + (c\theta c\varphi + s\psi s\varphi)\dfrac{U_1}{m} + W_1 \\ \ddot{\varphi} = -\dot{\theta}\dot{\psi}\left(\dfrac{I_z - I_y}{I_x}\right) - \dfrac{J_r}{I_x}\Omega_r\dot{\theta} - \dfrac{K_{fax}}{I_x}\dot{\varphi}^2 + \dfrac{U_2}{I_x} + W_2 \\ \ddot{\theta} = -\dot{\varphi}\dot{\psi}\left(\dfrac{I_z - I_x}{I_y}\right) + \dfrac{J_r}{I_y}\Omega_r\dot{\varphi} - \dfrac{K_{fay}}{I_x}\dot{\theta}^2 + \dfrac{U_3}{I_y} + W_3 \\ \ddot{\psi} = -\dot{\theta}\dot{\varphi}\left(\dfrac{I_y - I_x}{I_z}\right) - \dfrac{K_{faz}}{I_z}\dot{\psi}^2 + \dfrac{U_4}{I_z} + W_4 \end{cases} \tag{42}$$

where:

$$U = [U_1,\ U_2,\ U_3,\ U_4] = \begin{bmatrix} b\left(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2\right) \\ bl\left(\omega_4^2 - \omega_2^2\right) \\ bl\left(\omega_3^2 - \omega_1^2\right) \\ d\left(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2\right) \end{bmatrix} = \begin{bmatrix} u \\ \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \tag{43}$$

$W = [W_1,\ W_2,\ W_3,\ W_4]$ is the vector disturbances produced by the wind and is defined as:

$$W = \begin{bmatrix} \sum_{i=1}^{4} \dfrac{f_{vi}}{m} \\ \dfrac{(f_{v4} - f_{v2})\ l}{I_x} \\ \dfrac{(f_{v3} - f_{v1})\ l}{I_y} \\ \dfrac{\sum_{i=1}^{4} \tau_{traîné_i}}{I_z} \end{bmatrix} \tag{44}$$

These disturbances are applied to the quad-rotor in order to check the performances of the system control.

The main problem is that the quad-rotor should overcome these disturbances and must not be blown by the wind whatever its strength. Evidently, the relation between the weight of the quad-rotor and the speed wind must be taken into account and respected.

4. CONTROL STRATEGIES

In this section, a trajectory tracking hindrance and a position control problems are presented when the quad-rotor is evolving in a hostile environment.

The aim of the work is founded on two control techniques: a PD conventional control strategy and a PD-NN intelligent control strategy, a comparative study will be done based on the robustness of these controllers to the disturbances caused by the wind.

4.1. PD control strategies

The PD control strategy is presented in Fig. 4. Two types of PD controllers are adopted: the first one concerns positions and altitude control, the second one is opted for attitude angles control, and the PID controller is devoted to the motors control.

The classical PID controller is represented with its continuous transfer function as follows:

$$C(p) = K_P + \frac{K_I}{p} + K_D p \tag{45}$$

where: K_P is the proportional coefficient, K_I is integral coefficient and K_D is the derivative coefficient.

The control law expression is based on the error between the desired variable and the process variable and is given by:

$$u(t) = K_P e(t) + K_I \int e(t)dt + K_D \frac{de(t)}{dt} \tag{46}$$

The PD controllers for positions: x, y and z respectively are described as:

$$u_x = K_{Px}(x_{des} - x) + K_{Dx}\left(\dot{x}_{des} - \dot{x}\right) \tag{47}$$

$$u_y = K_{Py}\left(y_{des} - y\right) + K_{Dy}\left(\dot{y}_{des} - \dot{y}\right) \tag{48}$$

$$u_z = K_{Pz}(z_{des} - z) + K_{Dz}\left(\dot{z}_{des} - \dot{z}\right) \tag{49}$$

The PD controllers for angles (Altitude and Attitude):

Roll 'φ' , pitch 'θ'and yaw 'ψ' respectively are described as:

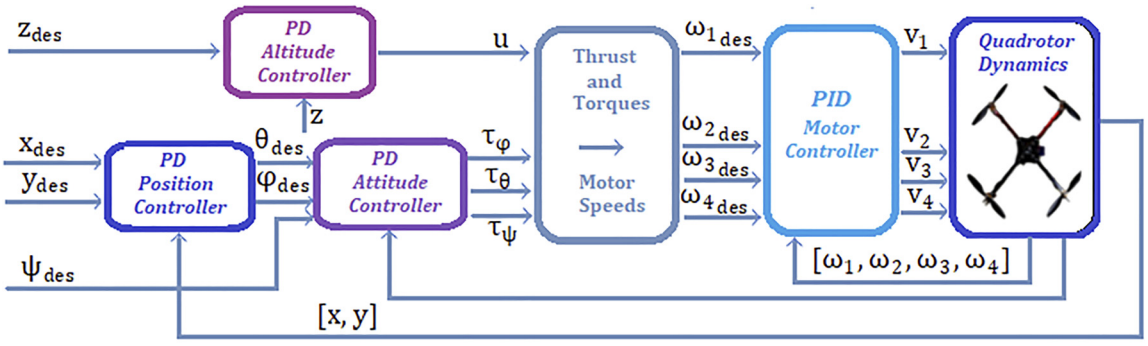

Fig. 4. PD and PD control strategies

$$u_{\varphi} = K_{P^{\varphi}}(\varphi_{des} - \varphi) + K_{D^{\varphi}}\left(\dot{\varphi}_{des} - \dot{\varphi}\right) \quad (50)$$

$$u_{\theta} = K_{P^{\theta}}(\theta_{des} - \theta) + K_{D^{\theta}}\left(\dot{\theta}_{des} - \dot{\theta}\right) \quad (51)$$

$$u_{\psi} = K_{P^{\psi}}(\psi_{des} - \psi) + K_{D^{\psi}}\left(\dot{\psi}_{des} - \dot{\psi}\right) \quad (52)$$

The PID controllers for motors speeds are respectively described as:

$$u_{\omega 1} = K_{P^{\omega 1}}(\omega_{1des} - \omega_1) + K_{I\omega 1}\int(\omega_{1des} - \omega_1)dt + K_{D^{\omega 1}}\left(\dot{\omega}_{1des} - \dot{\omega}_1\right) \quad (53)$$

$$u_{\omega 2} = K_{P^{\omega 2}}(\omega_{2des} - \omega_2) + K_{I\omega 2}\int(\omega_{2des} - \omega_2)dt + K_{D^{\omega 2}}(\dot{\omega}_{2des} - \dot{\omega}_2) \quad (54)$$

$$u_{\omega 3} = K_{P^{\omega 3}}(\omega_{3des} - \omega_3) + K_{I\omega 3}\int(\omega_{3des} - \omega_3)dt + K_{D^{\omega 3}}\left(\dot{\omega}_{3des} - \dot{\omega}_3\right) \quad (55)$$

$$u_{\omega 4} = K_{P^{\omega 4}}(\omega_{4des} - \omega_4) + K_{I\omega 4}\int(\omega_{4des} - \omega_4)dt + K_{D^{\omega 4}}\left(\dot{\omega}_{4des} - \dot{\omega}_4\right) \quad (56)$$

The tuning of the PID coefficients is based on the closed loop scheme with the use of the Ziegler and Nichols approach. The principle benefit of this method is its simplicity. It consists on setting the integral K_I gain to the extreme, and derivative K_D gain to zero.

Though, till it reaches the critical gain K_{osc}, the proportional K_P gain is increased, and the system oscillates continuously with T_{osc} period among constant amplitude oscillations. Formerly, the PID gains are set based on the gains K_{osc} and T_{osc} with mathematical approximations [27, 28].

Table 1. Numerical values of all PD and PID controllers

	K_P	K_I	K_D
Positions	5	0	8
Angles	9	0	14
Motors speeds	1	8	1

The numerical values of all PD and PID controllers are given in the next Table 1.

4.2. PID optimized neural networks approach: PID-NN control strategy

4.2.1. Neural network controller scheme. The main idea is the parameters optimization of the PD controller dedicated for the attitude angles, with the use of the famous feed-forward neural network. The methodology principle is to dynamically and on line adjust these parameters in order to reach the optimal PD-NN controller performances, according to the system running state.

The control strategy is based on three groups of controllers, one PD controllers group for the positions and the altitude, one PD-NN controllers group for the attitude angles and one PID controllers group for the speed of the quad-rotor.

The principle of the PD optimized neural networks approach for the attitude angles is shown in Fig. 5.

In this figure, the back propagation training algorithm is used to adjust the "P" and "D" gains related to the attitude angles. This method is based on the errors between the desired and the actual positions, angles and speeds.

The block diagram of the neural network is given in Fig. 6.

4.2.2. Neural network architecture. The neural networks used are multi-layer networks with the back-propagation training method. In this work, three neural networks where designed for attitude angles: Roll angle, Pitch angle and Yaw angle.

For each angle, as seen in Fig. 7, the structure of the neural network is composed of three layers. The input layer contains the actual and the desired angles, the hidden layer is composed of two hidden neurons that are the proportional function "P" and the derivative function "D" and the third layer deals with the torques corresponding to each angle [22].

The mathematical equations that govern the principle of these three similar neural networks are described in layers as follows:

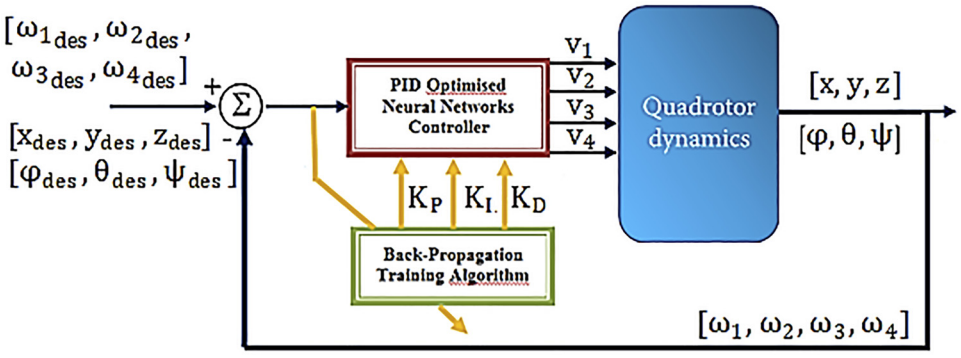

Fig. 5. Principle of PD optimized neural networks approach

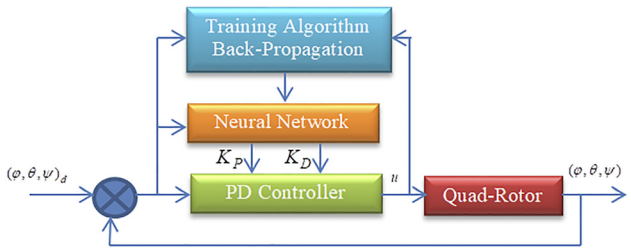

Fig. 6. Block diagram of the neural network approach

✓ The input layer with two input neurons:

$$\begin{cases} e_{1(\varphi,\theta,\psi)}(k) = (\varphi,\ \theta,\ \psi)(k) \\ e_{2(\varphi,\theta,\psi)}(k) = (\varphi,\ \theta,\ \psi)_d(k) \end{cases} \tag{57}$$

With k represents the iterations.

✓ The hidden layer composed of two hidden neurons and whose inputs are described by:

$$u_{j(\varphi,\theta,\psi)}(k) = \sum_{i=1}^{2} \omega_{ij(\varphi,\theta,\psi)}(k) e_{i(\varphi,\theta,\psi)}(k),\ j = 1,\ 2 \tag{58}$$

where $\omega_{ij}(k)$ are the weights between the hidden and the input layers.

So as:

$$\begin{cases} u_{1(\varphi,\theta,\psi)}(k) = \omega_{11(\varphi,\theta,\psi)}(k).e_{1(\varphi,\theta,\psi)}(k) + \omega_{21(\varphi,\theta,\psi)}(k).e_{2(\varphi,\theta,\psi)}(k) \\ u_{2(\varphi,\theta,\psi)}(k) = \omega_{12(\varphi,\theta,\psi)}(k).e_{1(\varphi,\theta,\psi)}(k) + \omega_{22(\varphi,\theta,\psi)}(k).e_{2(\varphi,\theta,\psi)}(k) \end{cases} \tag{59}$$

✓ The outputs of the hidden layer are described with:

$$\begin{cases} s_{1(\varphi,\theta,\psi)}(k) = u_{1(\varphi,\theta,\psi)}(k) = \omega_{11(\varphi,\theta,\psi)}(k) e_{1(\varphi,\theta,\psi)}(k) + \omega_{21(\varphi,\theta,\psi)}(k) e_{2(\varphi,\theta,\psi)}(k) \\ s_{2(\varphi,\theta,\psi)}(k) = \sum_{i=1}^{2} \omega_{i2(\varphi,\theta,\psi)} k) e_{i(\varphi,\theta,\psi)}(k) - \sum_{i=1}^{2} \omega_{i2(\varphi,\theta,\psi)}(k-1) e_{i(\varphi,\theta,\psi)}(k-1) \end{cases} \tag{60}$$

✓ The output layer has a single neuron whose main function is to provide the control signal. This neuron is expressed by:

$$\begin{aligned} (U_2, U_3, U_4)(k) = {} & \omega'_{1(\varphi,\theta,\psi)}(k).s_{1(\varphi,\theta,\psi)}(k) \\ & + \omega'_{2(\varphi,\theta,\psi)}(k).s_{2(\varphi,\theta,\psi)}(k) \end{aligned} \tag{61}$$

With ω'_j are the weights between the output and the hidden layers.

This gives:

$$\begin{aligned} (U_2,\ U_3,\ U_4)(k) = {} & \omega'_{1(\varphi,\theta,\psi)}(k) u_{1(\varphi,\theta,\psi)}(k) \\ & + \omega'_{2(\varphi,\theta,\psi)}(k)\left(u_{2(\varphi,\theta,\psi)}(k) - u_{2(\varphi,\theta,\psi)}(k-1)\right) \end{aligned} \tag{62}$$

Finally:

$$\begin{aligned} (U_2,\ U_3,\ U_4)(k) = {} & K_P\left(\omega_{11(\varphi,\theta,\psi)}(k) e_{1(\varphi,\theta,\psi)}(k)\right. \\ & \left. + \omega_{21(\varphi,\theta,\psi)}(k) e_{2(\varphi,\theta,\psi)}(k)\right) \\ & + K_D\left(\sum_{i=1}^{2} \omega_{i2(\varphi,\theta,\psi)}(k) e_{i(\varphi,\theta,\psi)}(k)\right. \\ & \left. - \sum_{i=1}^{2} \omega_{i2(\varphi,\theta,\psi)}(k-1) e_{i(\varphi,\theta,\psi)}(k-1)\right) \end{aligned} \tag{63}$$

Initially, the weights between the input layer and the hidden layer are chosen as such:

$$\begin{cases} \omega_{1j(\varphi,\theta,\psi)} = +1 \\ \omega_{2j(\varphi,\theta,\psi)} = -1 \end{cases} \quad j = 1, 2 \tag{64}$$

And the weights between the hidden layer and the output layer are defined, through the principle of the classical PD controller, by:

$$\begin{cases} \omega'_{1(\varphi,\theta,\psi)} = K_P \\ \omega'_{2(\varphi,\theta,\psi)} = K_D \end{cases} \tag{65}$$

The auto-tuning algorithm is the back-propagation training algorithm, because of its aptitude to adapt to changing environments. The back-propagation algorithm trains the neural network and updates the PD constants online. This algorithm to minimize the cost function E (k) by modifying the weights based on the gradient descent method. This cost function is expressed as the square of the difference between the input to the network and the desired input.

$$E(k) = \frac{1}{2} e^2_{(\varphi,\theta,\psi)}(k) \tag{66}$$

where $e_{(\varphi,\theta,\psi)}$ are the errors between the attitude angles and their desired values, and can be described as:

$$e_{(\varphi,\theta,\psi)}(k+1) = e_{(\varphi,\theta,\psi)}(k) + \Delta e_{(\varphi,\theta,\psi)}(k) \tag{67}$$

At first, the entries should be applied to the network; it propagates from the first layer to the hidden layers, to

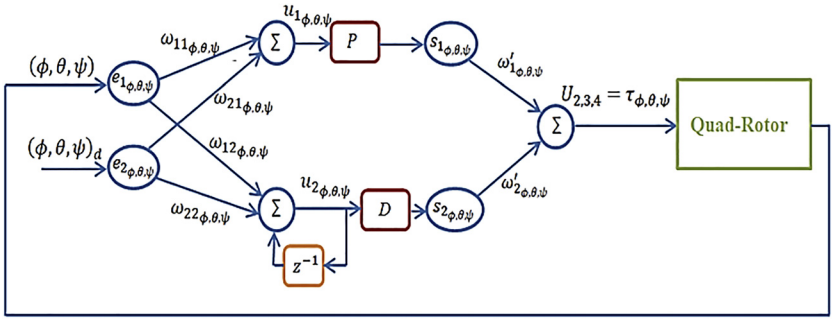

Fig. 7. Neural network structure for attitude angles control

output the angles of attitude. These angles are then compared to the desired values and a corresponding error vector is calculated. These errors propagate back, from the output layer, to all the neurons of the hidden layer so that all the neurons in the network have received an error signal thus involving it in its contribution to the total error.

The weights between the hidden layer and the output layer, which are the K_P and K_D coefficients of the PD controller, are updated by the following equation:

$$\omega'_j(k+1) = \omega'_j(k) - \eta_j.\Delta\omega'_j(k) \tag{68}$$

With: η_j is the learning rate and $\Delta\omega'_j(k)$ is the error gradient at unit j.

The variable quantity $\Delta e_{(\varphi,\theta,\psi)}(k)$ is defined as:

$$\Delta e_{(\varphi,\theta,\psi)}(k) = \frac{\partial e_{(\varphi,\theta,\psi)}(k)}{\partial\omega'_j(k)}.\Delta\omega'_j(k) \tag{69}$$

With this algorithm, the gradient is updated by the following expression:

$$\Delta\omega'_j(k) = -\eta\frac{\partial E(k)}{\partial\omega'_j(k)} = -\eta\frac{\partial E(k)}{\partial U_{2,3,4}(k)}.\frac{\partial U_{2,3,4}(k)}{\partial\omega'_j(k)} \tag{70}$$

Giving:

$$\Delta\omega'_j(k) = -e_{(\varphi,\theta,\psi)}(k).s_j(k).\delta(k) \tag{71}$$

where:

Table 2. Numerical optimized values of the PD-NN and the classical PD controllers for the attitude angles

Angles	K_P	K_D
Classical PD	10	20
PD-NN	12.5031	8.2124

$$\delta(k) = \mathrm{sgn}\frac{(\varphi,\ \theta,\ \psi)(k) - (\varphi,\ \theta,\ \psi)(k-1)}{U_{2,3,4}(k) - U_{2,3,4}(k-1)} \tag{72}$$

From the input layer to the hidden layer, the weights between the input layer and the hidden layer are updated by the following equation:

$$\omega_{ij}(k+1) = \omega_{ij}(k) - \eta_j.\Delta\omega_{ij}(k) \tag{73}$$

With: η_j is the learning rate and $\Delta\omega_{ij}(k)$ is the error gradient at unit j.

Where:

$$\Delta\omega_{ij}(k) = -e_{(\varphi,\theta,\psi)}(k).\dot{\omega_j}(k).e_i(k).\delta(k) \tag{74}$$

After the training algorithm, the optimized resultant numerical values of the PD-NN controller for the attitude angles are given in Table 2 and are compared with those of the classical PD.

5. SIMULATION RESULTS

5.1. Simulation results

- Controlled quad-rotor parameters responses with PD and PD-NN controllers

The training design of the PD and PD-NN controllers is established with the Matlab/Simulink environment, and simulation results are carried out with considering wind disturbances applied in all three directions. The control strategy consists of a desired trajectory which is founded on x_d, y_d and z_d desired positions and also on desired yaw angle $\psi_d = \sqrt{\mathrm{pi}/6}$. They are presented in Fig. 8.

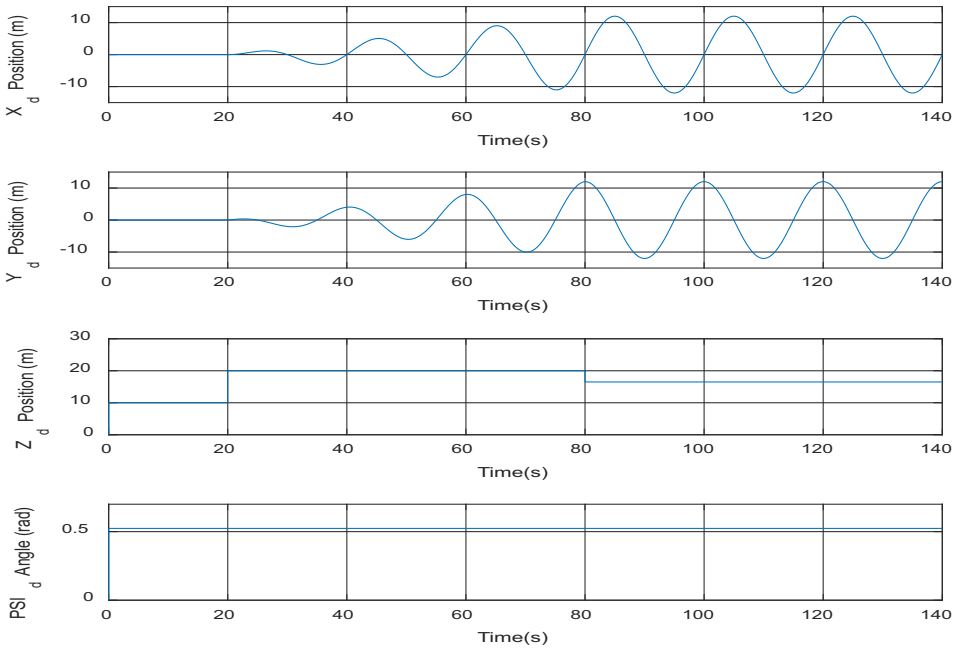

Fig. 8. x, y and z positions and ψ angle references

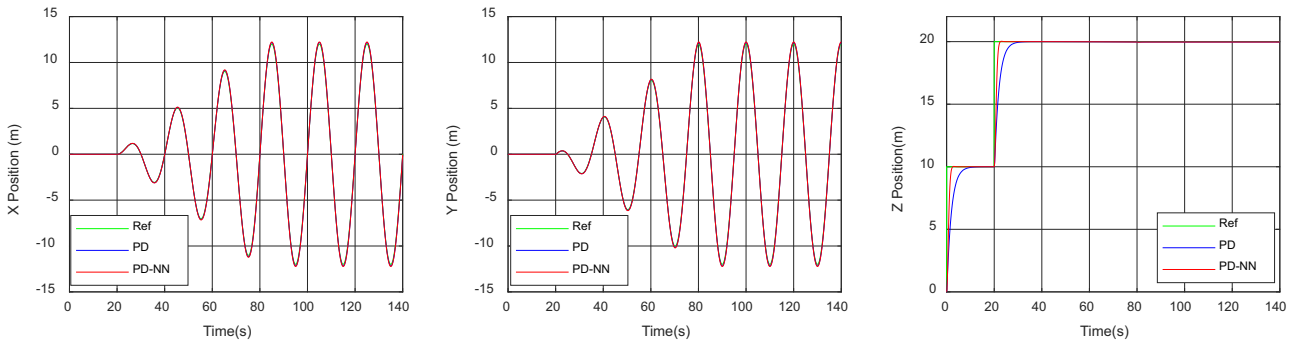

Fig. 9. Controlled x, y and z positions

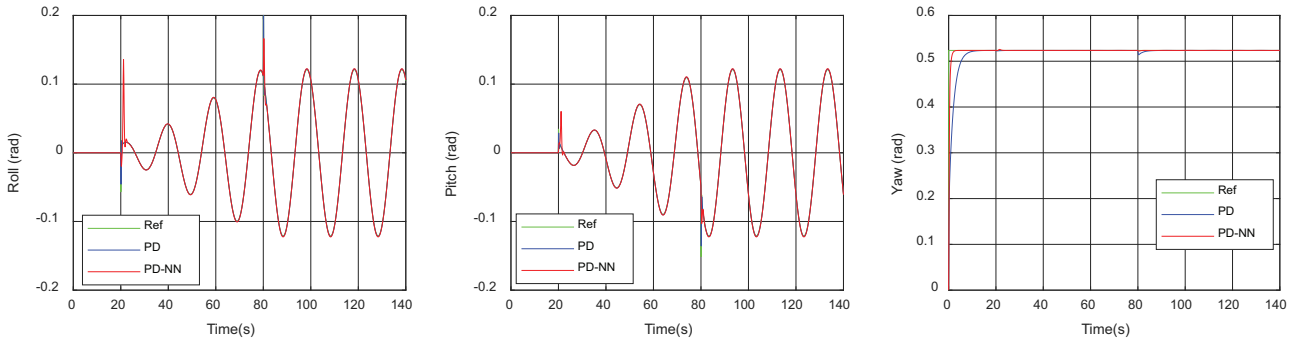

Fig. 10. Controlled roll, pitch and yaw angles

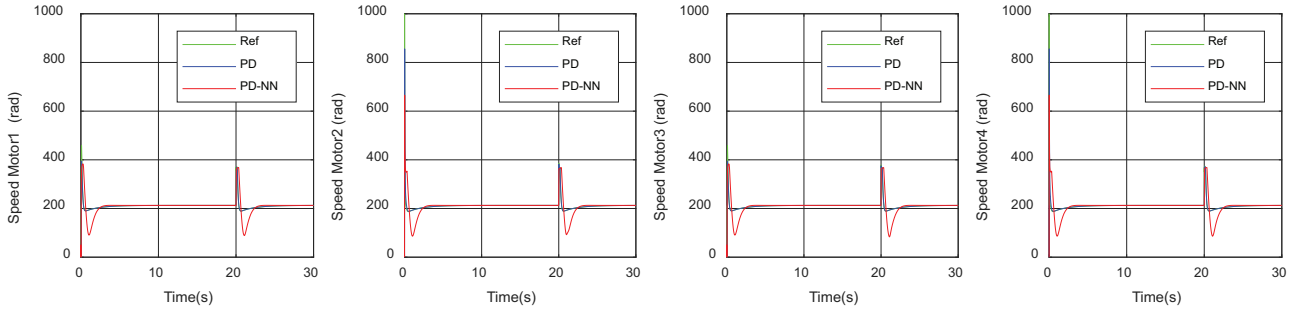

Fig. 11. Controlled motors speeds

The following Figs 9–12 represent results of all controlled quad-rotor parameters compared to references for both training control strategies PD and PD-NN controllers:

The trajectory tracking is given by:

- PD and PD-NN controllers comparative study with wind disturbances

The goal is to compare PD and PD-NN control strategies facing different levels of wind disturbances to test the robustness and the stability of both approaches against these troubles. Some wind velocities V_v were applied increasingly and two different stability thresholds were obtained for both control strategies. In fact, a comparative study is conducted and simulations results were obtained at what times several wind acceleration values were applied. The stability test is verified on the trajectory tracking and how the control strategy is able to let the quad-rotor following the reference trajectory.

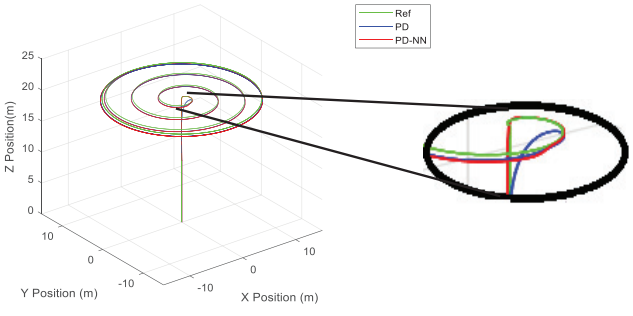

Fig. 12. Spiral trajectory of quad-rotor

Some simulations results are exposed for selected wind disturbances:

- $V_v = 3.24 \quad knots$

The trajectory tracking is represented by (Fig. 13):
$V_v = 9.18 \quad knots$
The trajectory tracking is represented by (Fig. 14):
Vv = 10:25 knots
The trajectory tracking is represented by (Fig. 15):
Vv = 21:06 knots
The trajectory tracking is represented by (Fig. 16):
To represent the variations of the quad-rotor positions with regards to the disturbances caused by the wind velocities, Fig. 17 highlights the stability thresholds of both control strategies.

- PD and PD-NN strategies for trajectories tracking

It is a task of varying the trajectories types in order to check the efficiency of both strategies to follow desired ones. A spiral trajectory is kept with rise and fall protocols.
Figure 18 shows the behavior of each control strategy:

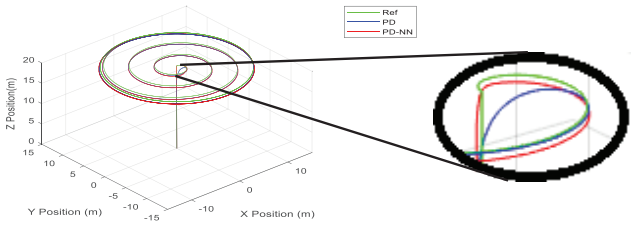

Fig. 13. Spiral trajectory of quad-rotor

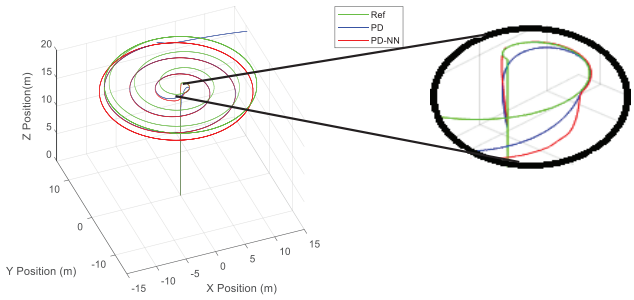

Fig. 14. Spiral trajectory of quad-rotor

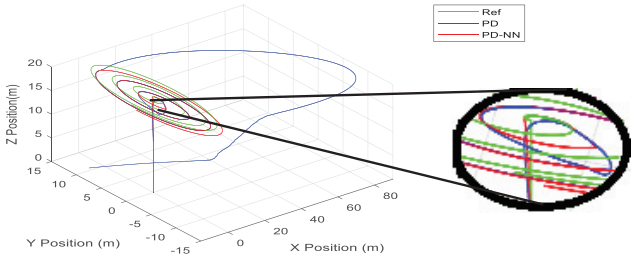

Fig. 15. Spiral trajectory of quad-rotor

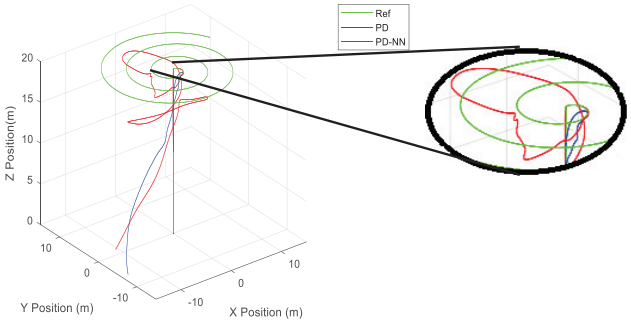

Fig. 16. Spiral trajectory of quad-rotor

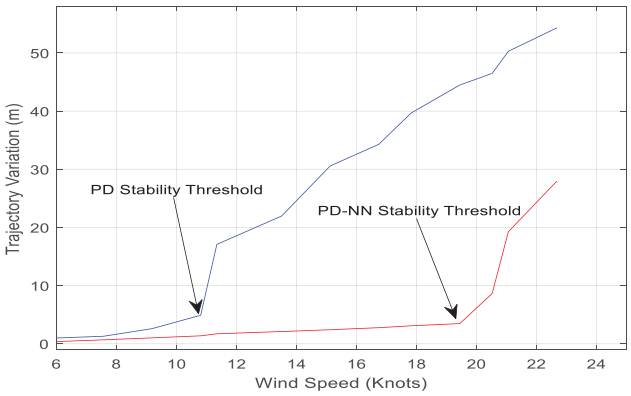

Fig. 17. Quad-rotor trajectory variations against wind disturbances

5.2. Experimental results with V-REP

The experimental results for the PD-NN controller are obtained based on the use of the V-REP environment with which some trajectories were considered and tested such as: circular, rectangular, eight and diamond shapes. These trajectories are represented in Fig. 19.

5.3. Experimental results with the BLADE Inductrix FPV+

The experimental results are applied on a quad-rotor BLADE Inductrix FPV+ described in Fig. 20.

The wind disturbances acting on the quad-rotor are applied with a fan and the trajectory to track is fixed as a rectangular shape.

Figure 21 shows the experimental result of trajectory tracking and it is observed that the quad-rotor manages to reach its trajectory.

6. DISCUSSION AND ANALYSES

In the previous section, desired performances were acted to on the quad-rotor and the controlled parameters responses were presented with regards to the variation of wind acceleration disturbances. Evidently, in all the previous cited cases, the quad-rotor should behave exactly like the references with both control strategies.

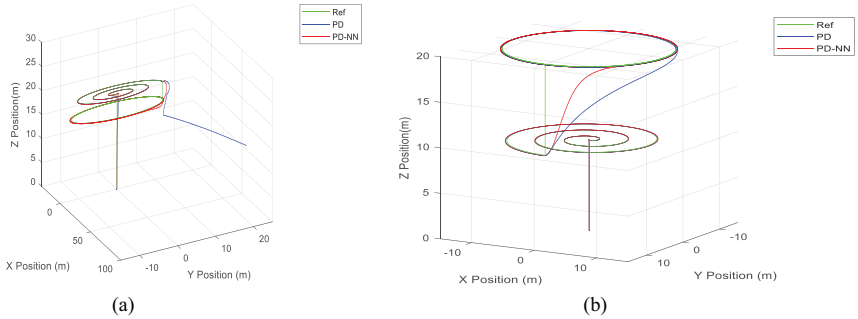

Fig. 18. (a) Quad-rotor medium fall from 20 m to 16 m, (b) quad-rotor very high rise from 10 m to 20 m

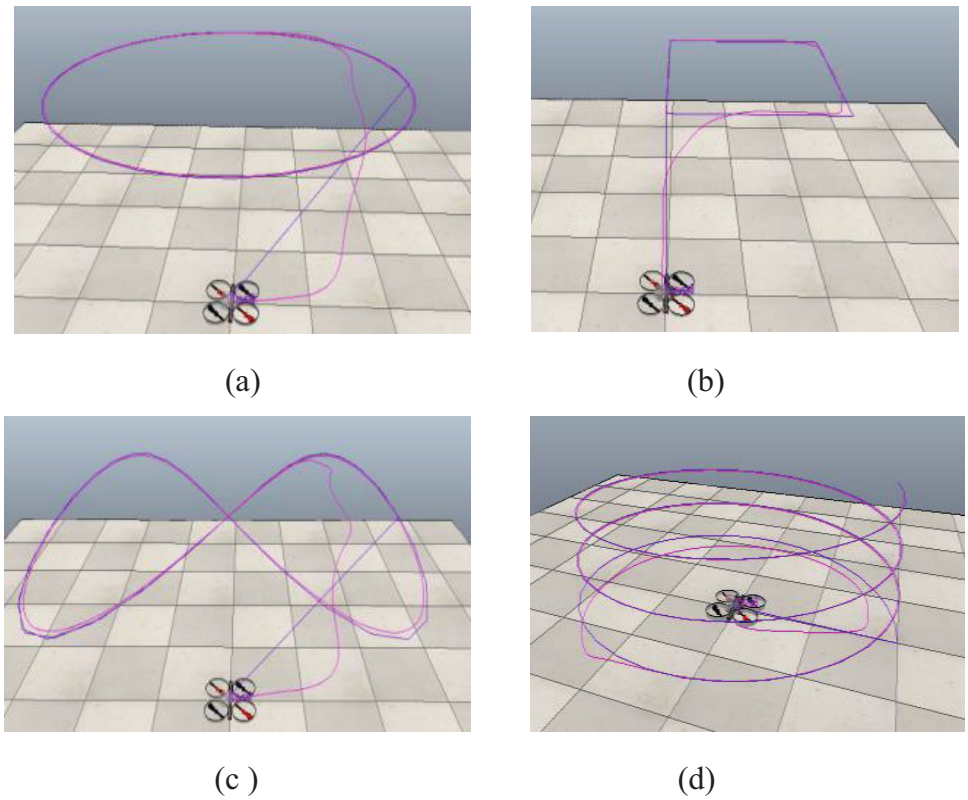

Fig. 19. (a) Circular shape, (b) rectangular shape, (c) eight shape and (d) diamond shape

Fig. 20. Quad-rotor BLADE Inductrix FPV+

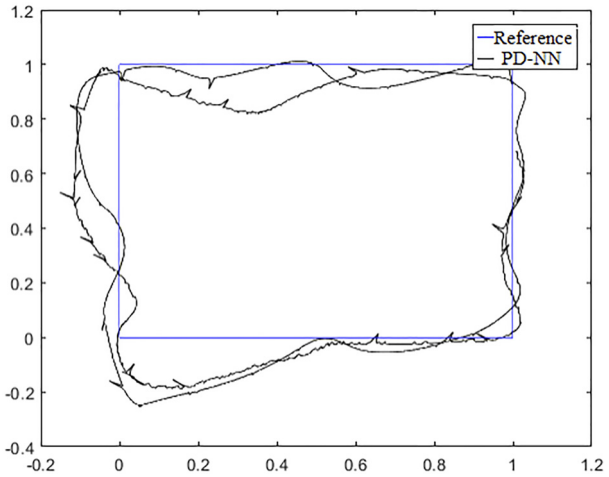

Fig. 21. Experimental results with rectangular trajectory

In fact, from Figs. 9–11, the quad-rotor follows the desired positions, angles and motors speeds with both controllers without adding wind disturbance.

In Fig. 12, the quad-rotor follows the desired trajectory. It is to confirm that the trajectory selected is very hard to track especially when the system is highly instable. In fact, this trajectory is composed of straight sections along z direction and compound alternations along x and y which lead to a circular and spiral trajectory.

In addition, it should be mentioned that the PD-NN controller offers faster and quick responses as represented in Fig. 22:

According to this figure, time responses at 5% are as follows:

For PID-NN controller $\tau_{r5\%} = 1.8s$

For PID controller $\tau_{r5\%} = 5.8s$

However, quad-rotor tests show fairly acceptable errors. Table 3 shows the maximum values of the errors reached for positions and angles for both strategies the classical one and the intelligent one.

The RMSE for the attitude angles between PD-NN and classical PD is given by Fig. 23:

When applying wind disturbances, the stability and the robustness of both control strategies were tested. Figures 13–16 shows the behavior of the proposed controllers facing wind disturbances. Effectively, when increasing disturbances, the controlled quad-rotor with PD and PD-NN controllers loses stability, however the second controller proposes better responses and the stability threshold is higher if compared with the PD controller.

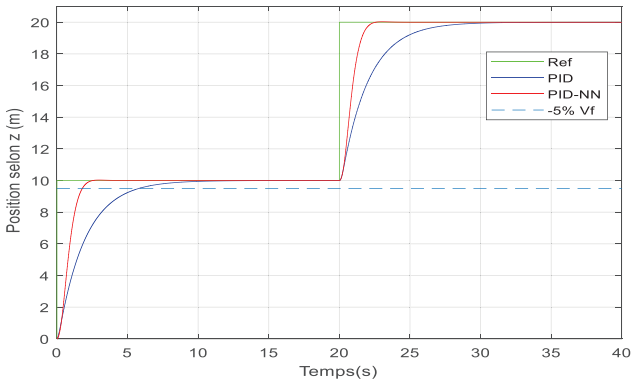

Fig. 22. PD and PD-NN controllers times responses at 5%

Figure 17 shows the trajectory variation against disturbance caused by the wind acceleration. It should be noticed that the controlled PD quad-rotor reaches instability from wind acceleration of 10.92 *knots*, but with the PD-NN, it reaches instability from wind acceleration of 19.65 *knots*. This value is considered to be acceptable and is tested in reality with the drone the BLADE Inductrix FPV+.

Concerning trajectory tracking, it is observed in Fig. 18, that the controlled PD quad-rotor cannot follow all types of trajectories in cases of falling down. Nevertheless, the controlled PD-NN quad-rotor can track all types of trajectories particularly with the fall and the rise.

Figure 19 shows that for the experimental results reached with the V-REP environment, the controlled PD-NN can follow all types of trajectories. Especially for the eight and diamond shapes that are more complicated than the others.

Figure 21 shows that for the experimental results reached with the BLADE Inductrix FPV+, the controlled PD-NN follows successfully the rectangular trajectories, mainly in presence of wind gust created through a fan with an air outlet speed of 17.49 *knots*.

Figure 23 represents the RMSE for the attitude angles between the classical PD and the PD-NN controllers and demonstrates the effectiveness of the PD-NN strategy over the classical one. The results of this test seems to be satisfactory in terms of difference between the two control strategies and demonstrates that the PD-NN controller is the best.

Basically, the PD-NN controller allows improved responses whether in the case of applying wind disturbances with different degrees: it seems to have a large margin of stability against these disturbances. Or in the case of trajectory tracking: it follows all the trajectories without any constraint. It might also be noted that in hover state and in the presence

Table 3. Errors values for positions and angles for both strategies

Strategy	x_{pos}	y_{pos}	z_{pos}	φ	θ	ψ
PID	0.2596	0.2597	0.0580	0.1287	0.0279	0.0096
PID-NN	0.2048	0.2047	0.0451	0.0569	0.0145	0.0014

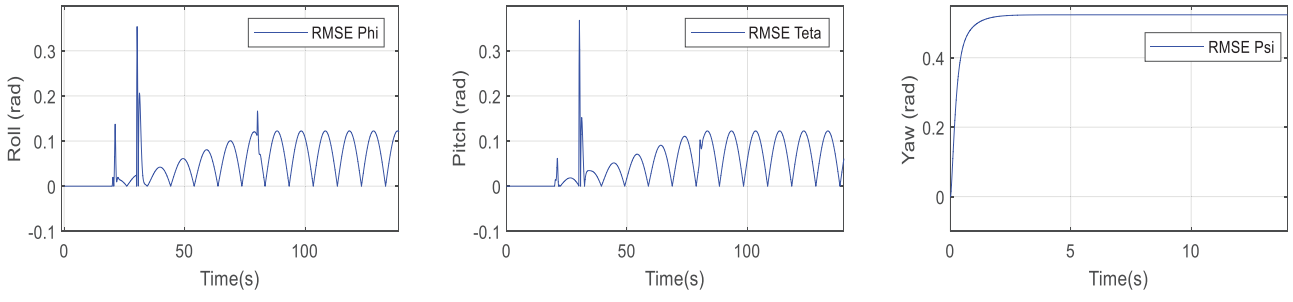

Fig. 23. RMSE for the attitude angles between PD-NN and classical PD

of wind disturbances, the quad-rotor can maintain its attitude. This case demonstrates its stability and robustness.

These results are compared with the work presented in [22], in which a linear active disturbance rejection control (LADRC) is proposed for stability control of an aerial robot quad-rotor under wind gusts. It is noticed that the controlled PID quad-rotor reaches instability from wind acceleration of 7, 77538 knots, but with the LADRC, it reaches instability from wind acceleration of 19, 4384 knots.. Basically, the LADRC and the PD-NN control strategies have strong ability of wind disturbance compensation and have comparable stability thresholds to wind gust disturbances.

7. CONCLUSION

In this paper, a PD and a PD-NN control schemes were proposed for a quad-rotor that is progressing in an environment with outdoor influences. The work treats the problem of wind turbulences and proposes a solution to this problem based on the PD-NN controller.

Moreover, the used trajectory is very difficult to follow, but the PD-NN controller optimized by neural networks leads to excellent results in terms of trajectory following and insensitivity to strong wind disturbances either with simulation or with experimentation.

In fact, it was revealed that this controller is robust beside different rigorous wind accelerations. It also promises stability and precision for static error performances. Simulation and experimental results of the PD-NN for the quad-rotor control strategy were given to verify and ensure the effectiveness of the controlled system in terms of quick and accurate responses.

Finally, it is to guarantee that the objective of this work was reached. Effectively, when progressing in a hostile environment with wind disturbances, the controlled quad-rotor is able to defend itself against these troubles without disturbing its trajectory following, its stability and its accuracy.

Basically, some other optimization algorithms are considered to be used in future works such as the Particle Swarm Optimization (PSO) algorithm, the Social Spider Optimization (SSO) algorithm and the Grey Wolf Optimization (GWO) algorithm which are useful tools to tune the parameters of proportional-derivative (PD) versions.

ABOUT THE AUTHORS

Chiraz Ben Jabeur holds a Master Degree (2002) and a PhD Degree in Electrical Engineering (2007). Now she is an Assistant Professor at the higher institute of informatics(ISI) in the Department of Electrical Engineering and Computer Science. She is a member of the research laboratory: Products Research Center (RIFTSI) at the ENSIT University of Tunis. Her current research deals with systems Control and artificial intelligence such as neural networks, fuzzy logic, genetic algorithms related mobile robots' domain.

Hassene Seddik is an associate professor in the ENSIT School. He has obtained the electromechanical engineering degree in 1995 and followed by the master degree in "signal processing: speaker recognition" and the thesis degree in data security and owner right privacy "watermarking using non-conventional transformations". He obtained the HDR degree equivalent to the ability to supervise researches leading to obtain the full professor degree in the field of intelligent data filtering and securing. His domain of interest is: data security Audio-image and video processing applied in intelligent filtering, encryption and watermarking.

REFERENCES

[1] G. Singhal, B. Bansod, and L. Mathew, "Unmanned aerial vehicle classification, applications and challenges: A review," *Preprints*, p. 19, 2018. Available: https://doi.org/10.20944/preprints201811.0601.v1. Accessed: Nov. 27, 2018.

[2] J. Ajmera and V. Sankaranarayanan, "Trajectory tracking control of a quad-rotor," in *International Conference on Control Communication & Computing India (ICCC)*, Trivandrum, 2015, pp. 48–53.

[3] C. Wang, W. Liu, and M. Q. Meng, "Obstacle avoidance for quad-rotor using improved method based on optical flow," in *IEEE International Conference on Information and Automation*, Lijiang, 2015, pp. 1674–9.

[4] T. Ustunkok and M. Karakaya, "Effect of PSO tuned P, PD, and PID controllers on the stability of a quad-rotor," in *1st International Informatics and Software Engineering Conference (UBMYK)*, Ankara, Turkey, 2019, pp. 1–6.

[5] J. J. Castillo-Zamora, K. A. Camarillo-GóMez, G. I. PéRez-Soto, and J. RodríGuez-ReséNdiz, "Comparison of PD, PID and sliding-mode position controllers for V–tail quadcopter stability," *IEEE Access*, vol. 6, pp. 38086–96, 2018. Available: https://doi.org/10.1109/ACCESS.2018.2851223. Accessed: Jun. 28, 2018.

[6] H. S. Khan and M. B. Kadri, "Attitude and altitude control of quad-rotor by discrete PID control and non-linear model predictive control," in *Paper presented at the International Conference on Information and Communication Technologies (ICICT)*, Karachi, 2015, pp. 1–11.

[7] H. Liu and G. Gao, "Dynamic modeling and analyzing for a novel X-quad-rotor," in *Paper presented at the 3rd International Conference on Computational Intelligence & Communication Technology (CICT)*, Ghaziabad, 2017, pp. 1–4.

[8] K. El Hamid, M. Mjahed, A. El Kari, and H. Ayad, "Neural and fuzzy based nonlinear flight control for an unmanned quad-rotor," *Paper Presented Int. Rev. Automatic Control (IREACO)*, vol. 11, no. 3, 2018.

[9] O. Doukhi, A. Fayjie, and D. Lee, "Intelligent controller design for quad-rotor stabilization in presence of parameter variations," *J. Adv. Transport.*, vol. 2017, p. 10, 2017.

[10] M. Algabri, H. Mathkour, M. A. Mekhtiche, M. A. Bencherif, M. Alsulaiman, M. A. Arafah, and H. Ghaleb, "Wireless vision-based fuzzy controllers for moving object tracking using a quad-rotor," *Int. J. Distributed Sensor Networks*, vol. 13, no. 4, 2017.

[11] H. Housny, E. Chater, and H. ElFadil, "Fuzzy PID control tuning design using Particle Swarm optimization algorithm for a quad-rotor," in *Paper presented at the IEEE 5th International Conference on Optimization and Applications (ICOA)*, 2019.

[12] J. Xu, Y. Ma, and J. Huang, "3D path tracking controller based on fuzzy PID optimized by PSO for quad-rotor," in *Chinese Automation Congress (CAC)*, 2019, pp. 3522–7.

[13] T. Liu, Y. Chen, Z. Chen, H. Wu, and L. Cheng, "Adaptive fuzzy fractional order PID control for 6-DOF quad-rotor," in *39th Chinese Control Conference (CCC)*, 2020, pp. 2158–63.

[14] S. Bari, S. S. Zehra Hamdani, H. U. Khan, M. U. Rehman, and H. Khan, "Artificial neural network based self-tuned PID controller for flight control of quadcopter," in *International Conference on Engineering and Emerging Technologies (ICEET)*, 2019, pp. 1–5.

[15] M. Fatan, B. L. Sefidgari, and A. V. Barenji, "An adaptive neuro PID for controlling the altitude of quadcopter robot," in *18th International Conference on Methods & Models in Automation & Robotics (MMAR)*, 2013, pp. 662–5.

[16] S. Furukawa, S. Kondo, A. Takanishi, and H. Lim, "Radial basis function neural network based PID control for quad-rotor flying robot," in *17th International Conference on Control, Automation Systems (ICCAS)*, 2017, pp. 580–4.

[17] H. C. T. E. Fernando, A. T. A. De Silva, M. D. C. De Zoysa, K. A. D. C. Dilshan, and S. R. Munasinghe, "Modelling, simulation and implementation of a quad-rotor UAV," in *IEEE 8th International Conference on Industrial and Information Systems*, 2013, pp. 207–12.

[18] G. Jithu and P. R. Jayasree, "Quad-rotor modelling and control," in *International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, 2016, pp. 1167–72.

[19] M. Walid, N. Slaheddine, A. Mohamed, and B. Lamjed, "Modelling, identification and control of a quad-rotor UAV," in *15th International Multi-Conference on Systems, Signals & Devices (SSD)*, 2018, pp. 1017–22.

[20] T. A. Ahmad, E. S. Fernando, A. K. Nashwa, K. Anis, A. Adel, J. H. Amjad, and K. I. Ibraheem, "Lagrangian dynamic model derivation and energy shaping control of non-holonomic unmanned aerial vehicles," in *Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2021)*, vol. 1377, 2021, pp. 483–93.

[21] M. Ardema, *Newton-Euler Dynamics*. Springer Science and Business Media, Inc, 2005.

[22] Y. Chen, Y. He, and M. Zhou, "Decentralized PID neural network control for a quad-rotor helicopter subjected to wind disturbance," *J. Cent. South Univ.*, vol. 22, no. 1, pp. 168–79, 2015.

[23] L. Ding and Z. Wang, "A robust control for an aerial robot quad-rotor under wind gusts," *J. Robotics*, pp. 1–8, 2018.

[24] G. Perozzi, D. Efimov, J. Biannic, L. Planckaert, and P. Coton, "Wind estimation algorithm for quad-rotors using detailed aerodynamic coefficients," in *Annual American Control Conference (ACC)*, 2018, pp. 1921–6.

[25] Y. Guo, B. Jiang, and Y. Zhang, "A novel robust attitude control for quad-rotor aircraft subject to actuator faults and wind gusts," *IEEE/CAA J. Automatica Sinica*, vol. 5, no. 1, pp. 292–300, 2018.

[26] R. T. Palomaki, N. T. Rose, M. Van Den Bossche, T. J. Sherman, and S. F. J. De Wekker, "Wind estimation in the lower atmosphere using multirotor aircraft," *J. Atmos. Oceanic Technol.*, vol. 34, no. 5, pp. 1183–91, 2017.

[27] Z. He and L. Zhao, "A simple attitude control of quad-rotor helicopter based on Ziegler-ichols rules for tuning PD parameters," *Sci. World J.*, vol. 2014, p. 13, 2014.

[28] K. J. Astrom and T. Hagglund, "Revisiting the Ziegler–Nichols step response method for PID control. Elsevier," *J. Process Control*, vol. 14, pp. 635–50, 2004.