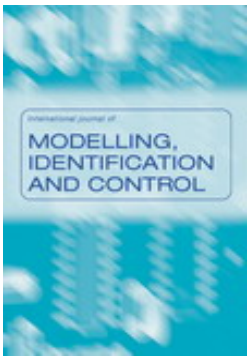


International Journal of Modelling, Identification and Control

ISSN online: 1746-6180 - ISSN print: 1746-6172
https://www.inderscience.com/ijmic

A new nonlinear PID controller design for a quadrotor system using teaching learning based optimisation algorithm

Naima Bouhabza, Kamel Kara

DOI: 10.1504/IJMIC.2023.10058548

Article History:

Received:	24 January 2023
Last revised:	03 April 2023
Accepted:	21 June 2023
Published online:	30 September 2024

A new nonlinear PID controller design for a quadrotor system using teaching learning based optimisation algorithm

Naima Bouhabza* and Kamel Kara

Labortoire des Systèmes Électriques et Télécommande,
Faculté de Technologie,
Université Blida1,
B.P 270, Route de Soumaa, Algeria
Email: naima.bouhabza@gmail.com
Email: km_kara@yahoo.fr
*Corresponding author

Abstract: In this paper, a novel nonlinear proportional integral derivative controller based on the meta-heuristic optimisation technique is suggested. Due to its straightforward implementation and structure, the proportional integral derivative controller is frequently employed in nonlinear system control. The Teaching_Learning_Based_Optimisation algorithm, owing due to its effectiveness, rapidity, and minimum initialisation parameter required, has gained the attention of a significant number of researchers. The quadrotor's actuation dynamics are controlled by nonlinear proportional integral derivative controllers. Moreover, under-actuated dynamics use the same controller mechanism. For each controller, six parameters are tuned using the integral time absolute error criteria. Through numerical simulation, the efficiency and control performance of the suggested scheme are proven and contrasted with those of the linear proportional integral derivative controller and the sliding mode control. The simulation research demonstrates the effectiveness and successful performance of the recommended control technique in terms of the transient response characteristics, tracking precision, and perturbation rejection.

Keywords: quadrotor; teaching learning based optimisation; TLBO; optimisation; nonlinear PID control.

Reference to this paper should be made as follows: Bouhabza, N. and Kara, K. (2024) 'A new nonlinear PID controller design for a quadrotor system using teaching learning based optimisation algorithm', *Int. J. Modelling, Identification and Control*, Vol. 45, No. 1, pp.1–10.

Biographical notes: Naima Bouhabza received her Licence Diploma in Aeronautic from the University of Blida, Algeria in 2014 and Master's Diploma in Automatic from the University of Blida, Algeria in 2016. Currently, she is a PhD student in Automatic, Department of Electronics, University of Blida, Algeria.

Kamel Kara received his Engineering Diploma in Electronics from the University of Setif, Algeria in 1992, Magister diploma in Electronics from the University of Constantine, Algeria in 1995, and Doctoral degree from the University of Setif, Algeria, in 2006. Currently, he is a Full Professor in Signal Processing and Systems Control at the Department of Electronics, University of Blida 1, Algeria. His current research interests are focused on nonlinear systems identification and control, artificial intelligence, heuristic optimisation, diagnostic and control of photovoltaic systems and embedded systems.

1 Introduction

The quadrotor is an unmanned aerial vehicle (UAV) which means aircraft that do not have pilots on board, with four rotors mounted in a cross shape, two of them rotate clockwise and the others anticlockwise. Using four motors and four propellers, a quadrotor can fly through the air and have the ability to take-off and land in limited space. The quadrotor, which is an unstable and under-actuated system, is typically used as a benchmark system to assess how well different control approaches perform. Applications for quadrotors include mapping, aerial cinematography, marine surveillance, combat support, traffic surveillance, and reconnaissance tasks for the military. Recently, research on quadrotor modelling and controlling have increased, and numerous strategies has been presented. Some of these approaches include: sliding mode control (SMC) (Ahmed and Chen, 2018; Hassani et al., 2022; Maqsood and Qu, 2020; Mofid et al., 2020; Nadda and Swarup, 2018), neural network control (Boudjedir, et al., 2012; Muliadi and Kusumoputro, 2018), fuzzy logic control (Erginer and

Altuğ, 2012; Ginting et al., 2022), predictive control (Abdolhosseini et al., 2013; Bangura and Mahony, 2014) and backstepping control (Rodríguez-Abreo et al., 2020; Yu et al., 2019).

It is worth mentioning that the most well-known and commonly used controller is the proportional integral derivative (PID) controller. The difficult side for a PID control scheme is the attainment of its best parameter values. In the literature, there are several techniques for obtaining the parameters of a PID controller; a review of these tuning methods has been given in Borase et al. (2021). Several meta-heuristic algorithms: techniques of application, equations and implementation flowcharts/ algorithms are provided in Joseph et al. (2022). Proportional derivative and PID controllers are optimised using the genetic algorithm (GA) in Siti et al. (2019). For the quadrotor's tracking path (Hasseni et al., 2021) explore the tuning of a quadrotor's PID controller's variables using stochastic nature-inspired algorithms; GA, evolution strategies, deferential evolutionary and cuckoo search. In Abbasi et al. (2013), the control of a quadrotor UAV has been considered using the PID control scheme with a fuzzy system to adjust the gains of the PID controllers. The combination of the PID control method with GA is used to control the altitude of a quadrotor in Khuwaja et al. (2018). A nonlinear PID controller is presented in Najm and Ibraheem (2019) to control the translational and rotational motions of the quadrotor; the parameters of the controllers are obtained using the GA. PID optimised by nature inspired algorithm, particle swarm optimisation (PSO), GA, and firefly algorithms (FA) have been explored in Moshayedi et al. (2020) to control the quadrotor system. (Alkamachi1 et al. (2017)) used PID controllers to stabilise the quadroter and attain the desired position and altitude; the variable values of the controller were determined using a GA. Extremum seeking algorithm (Muhsen and Raafat, 2021) is utilised to obtain better PID parameters values for controlling the quadrotor system. PSO and GA were used in Saribas and Kahvecioglu (2021) to tune fractional order PID controllers for quadrotor control. A PID controller based on PSO and Differential Evaluation to tune the parameters of PID has been implemented with real-time simulations of the quadrotor in Can and Ercan (2022). A quadrotor's attitude is controlled using online PID optimisation neural networks approach in Jabeur and Seddik (2022). A nonlinear fractional order PID (FOPID), which combines proportional gain with fractions of the derivative and integral actions to improve the exibility of the controller using PSO, has been suggested in Ibraheem and Ghusn (2016).

In the present work, the translational and rotational motions of the quadrotor system are controlled using a nonlinear structure of the well-known PID controller (NLPID). The position, altitude, and three orientation angles (roll, pitch, and yaw angles) of the quadrotor system are all controlled by six nonlinear PID controllers. There are six adjustable parameters for each controller. In actuality, the integral time absolute error (ITAE) criteria and the meta-heuristic optimisation algorithm 'teaching learning based optimisation (TLBO)' are applied to derive these values. The NLPID controllers have a nonlinear gain k whereas a linear PID controller has three parameters (k_p, k_i and k_d) that need to be optimised. The addition of this gain gives more design efficiency for the controller but significantly more difficult parameter setting. Despite the complexity and increased number of variables, this study shows the worth of the TLBO algorithm in controller tuning. Through numerical simulation, the effectiveness and control performance of the suggested scheme are proven and contrasted with those of the linear PID controller and the SMC. The simulation study demonstrates the effectiveness and successful performance of the recommended control technique in terms of the transient response characteristics, tracking precision, and disturbance rejection.

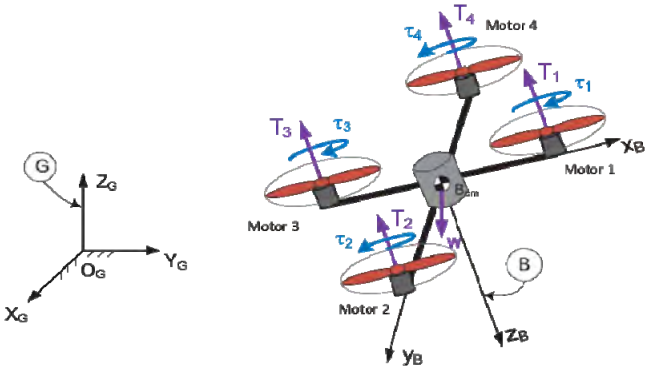
The remainder of the article is structured as follows. The quadrotor's mathematical model, which takes into account the motor's dynamic, is provided in Section 2. The TLBO algorithm is briefly discussed in Section 3, while the control's structure and algorithm are described in Section 4. Section 5 presents the simulation findings, while Section 6 draws some conclusions.

2 Quadrotor dynamic model

The quadrotor system is completely nonlinearly modelled using the Euler-Newton formalism before being employed in the development of the various controllers. An accurate quadrotor model may be created by taking into consideration the velocity and acceleration vectors. Figure 1 depicts a quadrotor with two motor pairs (1, 3) and (2, 4), each of which is capable of producing an independent thrust T_i, where i = 1, 2, 3, 4.

The four motors' speeds are changed to give the quadrotor its vertical motion to obtain the displacement along the X axis, increase or decrease the motors' (1) and (3) speeds (pitch motion) to accomplish the displacement along the Y axis, increase or decrease the motors' (2) and (4) speeds (roll motion). The yaw motion is caused by the variation in counter-torque between each pair of motors.

Figure 1 Quadrotor aircraft scheme (see online version for colours)



The dynamics of the quadrotor are described by Zeghlache et al. (2012) using the Newton Euler formalism, accounting for frictions caused by aerodynamic torques, drag forces, gyroscopic effects, and the rotors dynamic.

$$\begin{cases} \ddot{x} = \frac{1}{m}\left\{(U_x)U_1 - K_{ftx}\dot{x}\right\} \\ \ddot{y} = \frac{1}{m}\left\{(U_y)U_1 - K_{fty}\dot{y}\right\} \\ \ddot{z} = \frac{1}{m}\left\{(\cos\varphi\cos\theta)U_1 - K_{ftz}\dot{x}\right\} - g \\ \ddot{\varphi} = \frac{1}{I_x}\left\{\dot{\theta}\dot{\psi}\left(I_y - I_z\right) - K_{fax}\dot{\varphi}^2 - J_r\bar{\Omega}\dot{\theta} + dU_2\right\} \\ \ddot{\theta} = \frac{1}{I_y}\left\{\dot{\varphi}\dot{\psi}\left(I_z - I_y\right) - K_{fay}\dot{\theta}^2 + J_r\bar{\Omega}\dot{\varphi} + dU_3\right\} \\ \ddot{\psi} = \frac{1}{I_z}\left\{\dot{\theta}\dot{\varphi}\left(I_x - I_y\right) - K_{faz}\dot{\psi}^2 + bU_4\right\} \end{cases} \tag{1}$$

$$\begin{cases} U_x = \cos\varphi\sin\theta\cos\psi + \sin\varphi\sin\psi \\ U_y = \cos\varphi\sin\theta\sin\psi + \sin\varphi\cos\psi \end{cases} \tag{2}$$

U_i, i = 1, 2, 3, 4 the system control inputs are proportional to the angular speeds of the motors and are given by:

$$\begin{cases} U_1 = b\omega_1^2 + b\omega_2^2 + b\omega_3^2 + b\omega_4^2 \\ U_2 = b\omega_4^2 - b\omega_2^2 \\ U_3 = b\omega_3^2 - b\omega_1^2 \\ U_4 = d\left(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2\right) \\ \bar{\Omega} = \omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2 \end{cases} \tag{3}$$

From the quadrotor dynamic (1), the expressions of non-holonomic constraints are given by:

$$\begin{cases} \varphi_{de} = \arcsin\left(U_x\sin\psi_{de} - U_y\cos\psi_{de}\right) \\ \theta_{de} = \arcsin\left(\frac{U_x}{\cos\varphi\cos\psi_{de}} - \frac{\sin\varphi\sin\psi}{\cos\varphi\cos\psi_{de}}\right) \end{cases} \tag{4}$$

The motors dynamic is given by the following equations (Zeghlache et al., 2012)

$$\begin{cases} V = ri + L\frac{di}{dt} + k_e\omega \\ k_m = J_r + C_s + k_r\omega^2 \end{cases} \tag{5}$$

Then the model chosen for the rotors is as follows (Zeghlache et al., 2012):

$$\begin{cases} \dot{\omega}_i = k_m V_i - \beta_0 - \beta_1\omega_i - \beta_2\omega_i^2, \\ \beta_0 = \frac{C_s}{J_r},\ \beta_1 = \frac{k_e k_m}{rJ_r},\ \beta_2 = \frac{k_r}{J_r},\ \beta_0 = \frac{k_m}{rJ_r} \end{cases} \tag{6}$$

The description of each variable and each parameter of the model is given in Table 1.

Table 1 Description of the model variables and parameters

Parameters	*Description*
[x y z]	Linear position of the quadrotor
[φ θ ψ]	The angular position of the quadrotor
[ω_1, ω_2, ω_3, ω_4]	The motors speeds vector.
U_1	Vertical thrust
U_2	Pitching moment
U_3	Rolling moment
U_4	Yawing moment
$\bar{\Omega}$	The overall residual propeller angular speed.
m	Total mass
g	Gravitational force
I_x, I_y, I_z	Moment of inertia vector
K_{fax}, K_{fay}, K_{faz}	Aerodynamic friction coefficient
K_{ftx}, K_{fty}, K_{ftz}	Drag force coefficient
b	Thrust coefficient
d	Drag coefficient
V	Motor input
k_e	Electrical torque constant
k_m	Mechanical torque constant
k_r	Load constant torque
r	Motor internal resistance
L	Motor inductance
C_s	Solid friction
J_r	Motor inertia

3 TLBO algorithm

The TLBO algorithm, which measures the teacher's impact on the performance of students in the class, is based on the teaching-learning process (Rao and Patel, 2012; Rao et al., 2011, 2012). The algorithm identifies two fundamental forms of instruction:

1 learning from a teacher, 'teacher phase'

2 learning from other students 'learner phase'.

The many design variables in the optimisation problem are viewed as the various disciplines given to the student population. The learner's outcome is the optimisation problem's fitness value. The only two control elements needed for this strategy are the population size and the maximum number of repetitions.

Initialising the algorithm's parameters, which include the population size (n), the number of design variables (m), their upper and lower bounds (u, l), and the maximal number of repetitions $K_{\max}$, is the first stage. The following formula is used to produce the initial population:

$$X_{ij} = rand * (u_j - l_j) + l_j \quad i = 1, ..., n,\ j = 1, ..., m \tag{7}$$

where *rand* is a random number between 0 and 1.

For each iteration k (k = 1, ….., K_{max}), the following steps are taken during the teacher and learner phases:

3.1 Teacher phase

The teacher is considered as a very attentive individual who works to raise the mean result of the optimisation variables.

The mean result for all students in a certain subject is determined using:

$$M_j(k) = \sum_{i=1}^{n} \frac{X_{ij}(k)}{n}, \quad j = 1, ..., m \tag{8}$$

Then, a teacher $X_{bestj}(k)$ in a subject j is thought of as the variable that yields the fitness function's best value. Using the current mean result and the teacher's result for each subject, the different between the two is determined:

$$diff_j(k) = rand * (X_{bestj}(k) - T_f M_f(k)), \quad j = 1, ..., m \tag{9}$$

where T_f, which can be either 1 or 2, is the teaching factor, is given by:

$$T_f = round(rand + 1) \tag{10}$$

After, new solutions $X_{ij}^{new}(k)$ must be generated according to the equation below:

$$X_{ij}^{new}(k) = X_{ij}(k) + diff_j(k) \quad i = 1, ..., n \quad j = 1, ..., m \tag{11}$$

The new solution $X_{ij}^{new}(k)$ is accepted and becomes the existing solution $X_{ij}(k)$, if it yields the optimal fitness function value; else, the current solution must be maintained.

3.2 Learners phase

The learners interact among themselves in order to improve their knowledge throughout this phase. Two learners $X_{Pj}(k)$ and $X_{Qj}(k)$ are randomly selected to interact among themselves.

If $X_{Pj}(k)$ yields the optimal fitness function value than $X_{Qj}(k)$, the following equation produces a new solution, $X_{Pj}^{new}(k)$:

$$X_{Pj}^{new}(k) = X_{Pj}(k) + rand * (X_{Pj}(k) - X_{Qj}(k)) \tag{12}$$

Otherwise:

$$X_{Pj}^{new}(k) = X_{Pj}(k) + rand * (X_{Qj}(k) - X_{Pj}(k)) \tag{13}$$

If the new solution $X_{Pj}^{new}(k)$ gives the best value of the fitness function, it is approved and takes the place of the existing solution. In other, the present solution is preserved.

4 Nonlinear PID controllers design

4.1 Control structure

The quadrotor system's horizontal motions (x, y), altitude (z), roll angle (φ), pitch angle (θ) and yaw angle (ψ), are all controlled by six nonlinear PID controllers. The inner loop control and the outer loop control are the two control loops that build up the control structure in Figure 2.

In order to obtain a fast response with no overshoot a nonlinear gain k is added to the conventional linear PID controller.

The equations of the controllers are given as follows (Seraji, 1998):

$$\begin{cases} U_i = \left(k_p^i * e_i + k_i^i * \int e_i + k_d^i * \dfrac{de_i}{dt} \right) * k \\ k = \left(k_0^i + k_1^i * \left(\dfrac{2}{1 + \exp(-k_2^i * e_i - 1)} \right) \right) \end{cases} \tag{14}$$

where: i = x, y, z, φ, θ, ψ, $\left[k_p^i, k_i^i, k_d^i, k_0^i, k_1^i, k_2^i \right]$: are the NLPID controllers' parameters.

The error signals are given by:

$$\begin{cases} e_x = x_d - x_m \\ e_y = y_d - y_m \\ e_z = z_d - z_m \\ e_\varphi = \varphi_d - \varphi_m \\ e_\theta = \theta_d - \theta_m \\ e_\psi = \psi_d - \psi_m \end{cases} \tag{15}$$

The required reference paths for the horizontal position, altitude, roll, pitch, and yaw angles are, respectively: x_d, y_d, z_d, φ_d, θ_d, ψ_d. The related measured signals are x_m, y_m, z_m, φ_m, θ_m, ψ_m.

4.2 Control algorithm

The parameters values of the NLPID controllers are determined through minimising the ITAE criterion using the TLBO algorithm. The ITAE criterion is:

$$J = \sum_{i=1}^{6} ITAE_i \qquad ITAE_i = \int_0^\infty t * |e_i(t)| dt \tag{16}$$

Considering s = 1, …, n, j = 1, …, m, k = 1, …, K_{max}, n is the population size and m is the number of parameters that need to be adjusted; there are 36 parameters in this case, which are the gains of the six NLPID controllers.

The optimisation algorithm's maximum number of repetitions is K_{max}.

The fitness function is the criterion J and the used population is given by:

$$\boldsymbol{X} = \begin{bmatrix} X_1 \\ X_2 \\ . \\ . \\ . \\ X_n \end{bmatrix} \tag{17}$$

where each element of the matrix $\boldsymbol{X}$ is defined as follows:

$$\begin{aligned} X_S = \Big[& k_{p_s}^x k_{d_s}^x k_{i_s}^x k_{0_s}^x k_{1_s}^x \; k_{2_s}^x k_{p_s}^y k_{d_s}^y k_{i_s}^y k_{0_s}^y k_{1_s}^y k_{2_s}^y \; k_{p_s}^z \ldots \\ & \ldots k_{d_s}^z k_{i_s}^z k_{0_s}^z k_{1_s}^z k_{2_s}^z \; k_{p_s}^\varphi k_{d_s}^\varphi k_{i_s}^\varphi k_{0_s}^\varphi k_{1_s}^\varphi k_{2_s}^\varphi k_{p_s}^\varphi k_{d_s}^\varphi \ldots \\ & \ldots k_{i_s}^\theta k_{0_s}^\theta k_{1_s}^\theta k_{2_s}^\theta \; k_{p_s}^\psi k_{d_s}^\psi k_{i_s}^\psi k_{0_s}^\psi k_{1_s}^\psi k_{2_s}^\psi \Big] \\ & s = 1, ..., n \end{aligned} \tag{18}$$

To each element X_s of the population X is associated a fitness function J_s defined by (16). The following is a summary of each step in the optimisation algorithm:

Step 0 Select the size of the population (n) and the maximum number of repetitions $K_{\max}$, set the number of the optimisation parameters (m = 36). Specify the upper and lower values of each optimisation parameter (u_j, l_j, j = 1, …, m) and the different reference trajectories.

Step 1 Set the iterations index k = 1 and generate a random population $\boldsymbol{X}(\boldsymbol{k})$ using (7).

Step 2 Compute the corresponding control laws $U_i(k)$, $i = x, y, z, \varphi, \theta, \psi$ using the current population $\boldsymbol{X}(\boldsymbol{k})$ and (14).

Step 3 Determine the system responses to the computed control signals and compute the corresponding errors.

Step 4 For each element $X_s(k)$ of the population $\boldsymbol{X}(\boldsymbol{k})$, calculate the corresponding value of the fitness function $J_s(X_s(k))$ using (16). The element $X_h(k)$ with the minimum value $J_h(X_h(k))$, $h \in \{1, 2, 3, \ldots, n\}$ of the fitness function is taken as a teacher.

$X_{best}(k) = X_h(k)$ such that:

$$J_h\left(X_h(k)\right) \le J_s\left(X_s(k)\right), \quad h \in \{1, 2, ..., n\}, s = 1, ..., n$$

Step 5 Compute the mean result of the learners $M_s(X_s(k))$ using (8), the difference values $diff_s(k)$ using (9) and the teaching factor T_f using (10).

Step 6 Generate the new population $\boldsymbol{X}^{new}(k)$ according to (11).

Step 7 Compute the corresponding control laws $U_i(k)$, $i = x, y, z, \varphi, \theta, \psi$ using the new population $\boldsymbol{X}^{new}(k)$ and (14).

Then, determine the system responses to the computed control signals and compute the corresponding errors.

Step 8 Evaluate the fitness $J_s(X_s^{new}(k))$ of each element of $\boldsymbol{X}^{new}(k)$.

If a new solution $X_s^{new}(k)$, s = 1, …, n, gives the small value of the fitness function than the gerated solutions $X_i(k)$, i = 1, …, n, if so, it is accepted and the present solution is maintained; otherwise, it is not.

Step 9 Randomly choose two solutions $X_P(k)$ and $X_Q(k)$, $P, Q \in \{1, 2, \ldots, n\}$, $P \neq Q$, and generate new solution $X_s^{new}(k)$ according to (12) and (13).

Step 10 Compute the corresponding control laws $U_i(k)$, $i = x, y, z, \varphi, \theta, \psi$, using the new solution $X_s^{new}(k)$ and (14). Then, determine the system responses to the computed control signals and compute the corresponding errors.

Step 11 Evaluate the fitness $J_P(X_s^{new}(k))$.

If $J_P(X_s^{new}(k)) < J_Q(X_Q(k))$, the current solution is replaced by the new solution $X_s^{new}(k)$. Otherwise, the present solution is retained.

Step 12 The best solution is $X_{best}(k) = X_h(k)$ such that:

$$J_h\left(X_h(k)\right) \le J_s\left(X_s(k)\right), \quad h \in \{1, 2, ..., n\}, s = 1, ..., n$$

Step13 Increment the iteration index $k = k + 1$. If $k < K_{\max}$ go to step 5. Otherwise, the best solution corresponding to the current sampling time is that of step 12.

For the next sampling time the current population is taken as an initial population and the algorithm is started from step 1.

5 Simulation results

The MATLAB/Simulink environment is used to implement both the control algorithm and the quadrotor model that were previously presented. Table 2 compiles the quadrotor model's parameters for use in the simulation (Zeghlache et al., 2012).

The parameters values of the TLBO algorithm are listed in Table 3.

The obtained gains of the six NLPID controllers are given in Table 4.

Figure 2 Control structure of the quadrotor (see online version for colours)

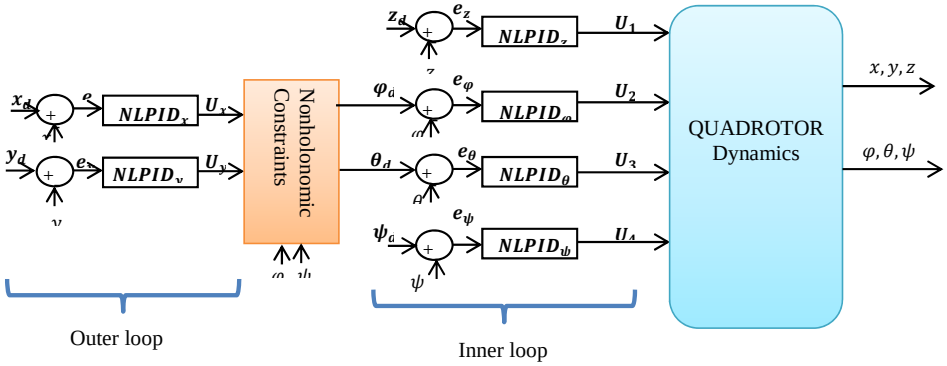


Table 2 Parameters of the quadrotor model

Parameter	*Value*
m	0.486 kg
g	9.8 m/sec^2
I_x, I_y, I_z	(3.8278, 3.8288, 7.6566) * 10^{-3} *N.m/rad/s*^{2}
K_{fax}, K_{fay}, K_{faz}	(5.5670, 5.5670, 6.3540) * 10^{-4} *N.m/rad/s*^{2}
K_{ftx}, K_{fty}, K_{ftz}	(5.5670, 5.5670, 6.3540) * 10^{-4} *N.m/rad/s*^{2}
b	2.9842 * 10^{-5} *N.m/rad/s*
d	2.2320 * 10^{-7} *N.m/rad/s*
J_r	2.8385 * 10^{-5} *N.m/rad/s*^{2}
β_0	189.63
β_1	6.0612
β_2	0.0122
b_m	280.19

Table 3 Parameters values of the TLBO algorithm

Parameter	*Value*
n	10
m	36
[l u]	[0 100]
$K_{\max}$	40

Table 4 Gains values of the NLPID controllers using the TLBO algorithm

	x	y	z	φ	θ	ψ
k_p	21.7752	8.1953	80.1916	59.0700	29.4263	7.3223
k_d	20.4921	10.5998	11.0120	32.6616	9.7013	3.4884
k_i	0.6198	0.0042	2.5476	0.1384	52.7631	0.0009
k_0	0.9649	0.5526	0.5845	0.4314	0.6381	0.0077
k_1	0.0603	0.0272	0.3670	0.8430	0.2238	0.0271
k_2	0.0927	0.0115	0.5614	0.2218	0.3225	0.0076

The effectiveness of the NLPID control strategy is compared with that of the optimised PID controller using TLBO (Bouhabza et al., 2021), and the SMC method presented in Zeghlache et al. (2012).

5.1 Case of step reference trajectories

In this section, a unit step as a desired trajectory is considered for the quadrotor's horizontal positions, altitude, and zero for yaw angle. The simulation is run with a sampling time of T_s = 0.001 s and the following input limitations: $0 \le U_1 \le 10$, $0.01 \le U_2 \le -0.01$, $0.01 \le U_3 \le -0.01$, $0.01 \le U_4 \le -0.01$. For (x, y, z, φ, θ, ψ), the starting conditions are (0, 0, 0, 0, 0, –0.5). Figures 3–8 display the outcomes employing the aforementioned control strategies. The NLPID control system achieves good control performance in terms of tracking accuracy and reaction speed with low overshoot, as illustrated in these figures.

Figure 3 Horizontal position x of the quadrotor (see online version for colours)

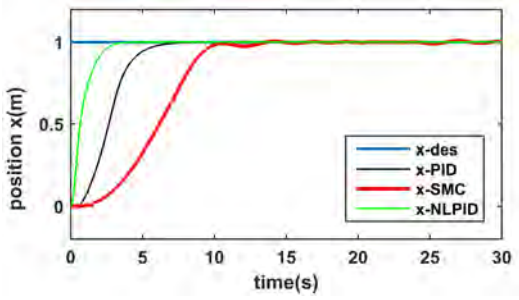


Figure 4 Horizontal position y of the quadrotor (see online version for colours)

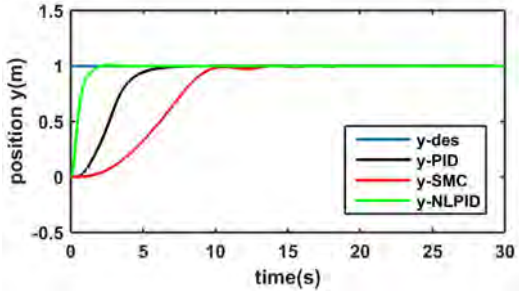


Figure 5 Altitude z of the quadrotor (see online version for colours)

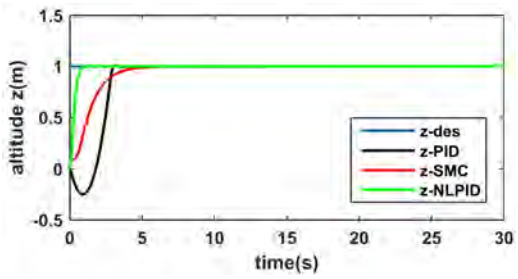


Figure 6 Roll (φ) angle of the quadrotor (see online version for colours)

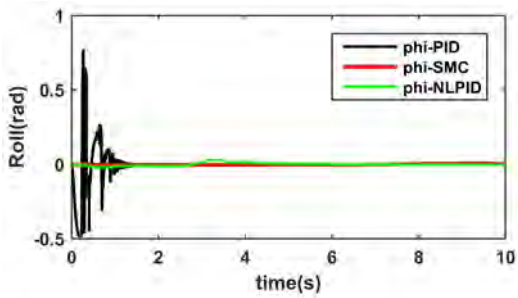


Figure 7 Pitch (θ) angles of the quadrotor (see online version for colours)

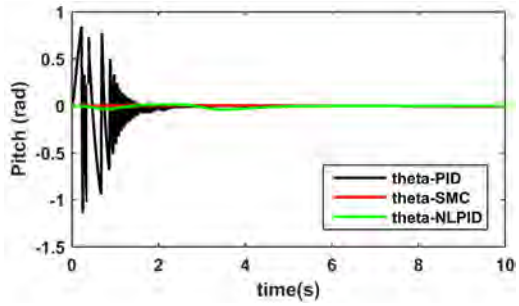
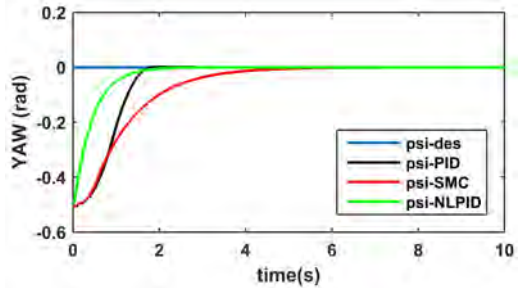


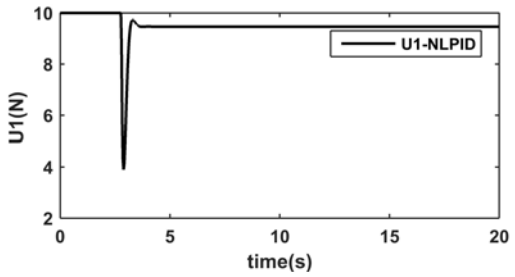
Figure 8 Yaw angle (ψ) of the quadrotor (see online version for colours)



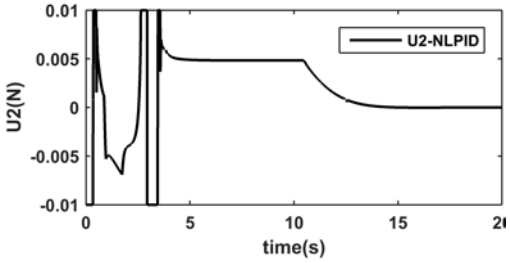
The control signal generated by the controllers to obtain the necessary reference is shown in Figure 9.

The ITAE criterion values for the three controllers are shown in Table 5. It can be seen that the smallest values of the ITAE criterion are obtained in the case of the NLPID controllers.

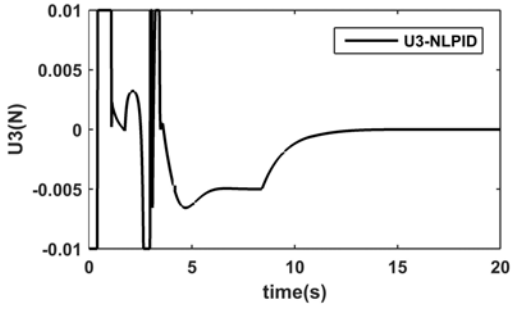
Figure 9 Control signals U_1, U_2, U_3 and U_4



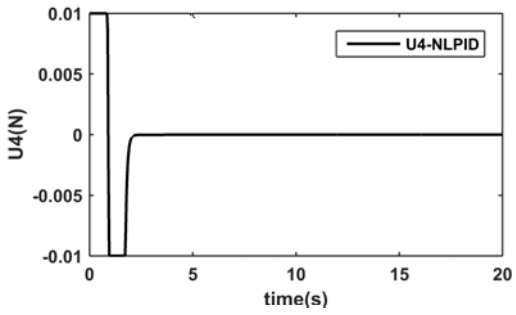
(a)



(b)



(c)



(d)

Table 5 Values of the ITAE criterion

	ITAE		
	NLPID	*PID*	*SMC*
x	4.116	47.452	211.709
y	4.116	47.425	211.710
z	2.964	35.203	16.938
φ	1.074	6.098	8.292
θ	1.310	9.845	8.315
ψ	2.626	3.277	7.243

Table 6 Transient responses characteristics for (x, y, z, ψ)

	NLPID			*PID*			*SMC*		
	$t_r(s)$	$t_s(s)$	$M_P(\%)$	$t_r(s)$	$t_s(s)$	$M_P(\%)$	$t_r(s)$	$t_s(s)$	$M_P(\%)$
x	1.0913	3.7752	0.1572	3.1619	6.0849	19.4395	5.7878	12.6285	0.9055
y	1.1011	1.9488	0.1572	3.1619	6.0849	1.4046	5.7828	12.6486	0.3899
z	0.4940	0.8076	0.1314	0.8804	2.9368	0.7294	2.5096	4.4992	0
ψ	0.9571	1.6076	0	0.8835	1.6542	0	2.2425	4.2985	0

For the three considered controllers, the values of the different characteristics of the transient responses (rising time t_r and settling time t_s and overshoot peak M_P) are gathered in Table 6.

5.2 *Measurement noise sensitivity and robustness against disturbances*

5.2.1 *Measurement noise sensitivity*

The effectiveness of a suggested approach must be studied in relation to measurement noise. We use a unit step to get the desired trajectories for x_d, y_d, z_d, and zero for ψ_d. To approximate sensor noise, the observed variables (x, y, z, φ, θ, ψ) are blended with white Gaussian noise (with zero mean and 10^{-5} variance). Figures 7, 8, and 9 exhibit the simulation findings. It is clear that the NLPID control strategy is less sensible than the PID control and the SMC.

Figure 10 Tracking results with measurement noise for x-position of the quadrotor (see online version for colours)

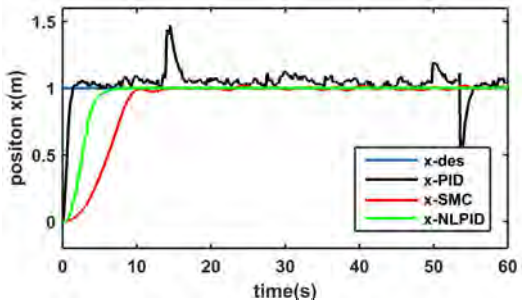


Figure 11 Tracking results with measurement noise for y-position of the quadrotor (see online version for colours)

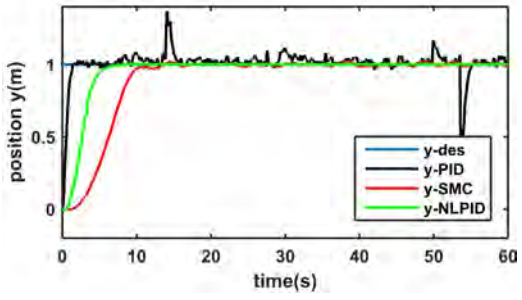


Figure 12 Tracking results with measurement noise for the altitude z of the quadrotor (see online version for colours)

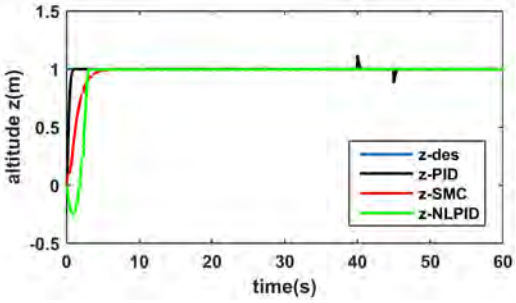


Figure 13 Tracking results with measurement noise for phi angle φ of the quadrotor (see online version for colours)

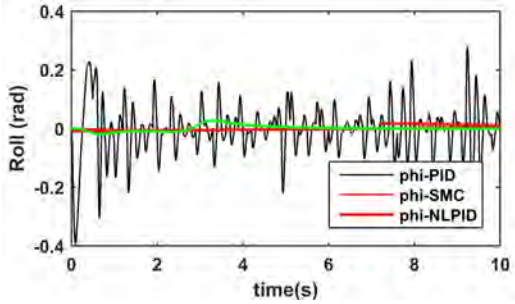
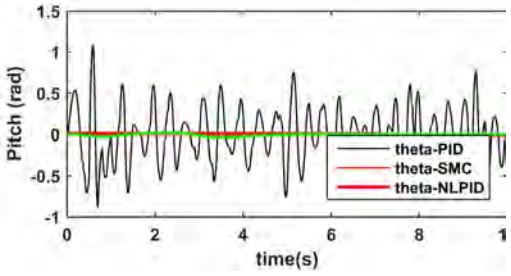


Figure 14 Tracking results with measurement noise for theta angle θ of the quadrotor (see online version for colours)



5.2.2 *Robustness against disturbances*

It is supposed that (x, y) and (z) of the quadrotor are exposed to additive perturbations of amplitude 0.12 m during [40 s 45 s] in order to evaluate the sensitivity of the recommended control method to external disturbances. Figures 16, 17, and 18 illustrate the obtained results, which demonstrate that the suggested control technique has a

minimal overshoot peak and can rapidly counteract the disturbance impact.

Figure 15 Tracking results with measurement noise for yaw angle ψ of the quadrotor (see online version for colours)

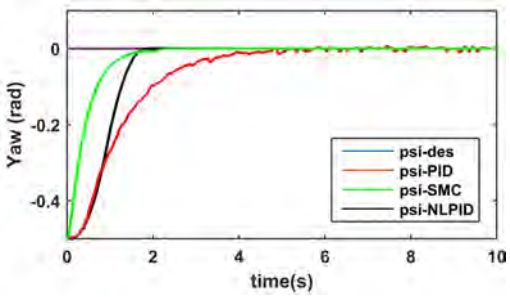


Figure 16 Tracking results with external disturbances for x-position of the quadrotor (see online version for colours)

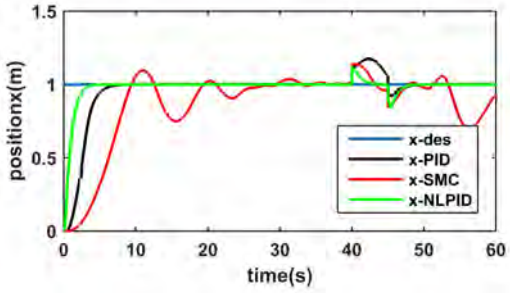


Figure 17 Tracking results with external disturbances for y-position of the quadrotor (see online version for colours)

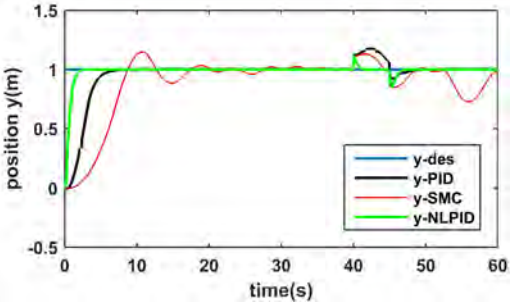
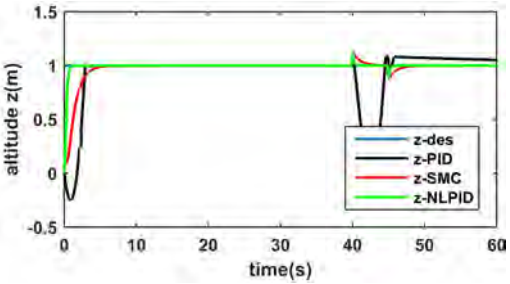


Figure 18 Tracking results with external disturbances for the altitude z of the quadrotor (see online version for colours)



6 Conclusions

In this paper, effective approaches have been developed to control the quadrotor system's translational and rotational movements. The complete nonlinear model of the quadrotor, including the dynamics of the motors, has been used to develop and verify the controllers' performance. The parameter values of the recommended control technique are determined using the TLBO algorithm and the ITAE criterion, and the control strategy combines a conventional linear PID controller with a nonlinear gain. In fact, the efficiency of this meta-heuristic optimisation method, that requires only the size of the population and the number of generations, has been proven in many engineering problems. The suggested control technique is relatively easy to implement and does not necessitate a lot of calculation time. The obtained results have demonstrated that the proposed control approach is capable to make the quadrotor system track the desired trajectories, stabilise the attitude angles and reject the disturbances. Obviously, the developed control algorithm can be used to control with success other nonlinear systems. In the near future work, we'll examine how well an actual quadrotor system performs utilising the active disturbance rejection control paradigm to actively reject the wind disturbances.

References

Abbasi, E., Mahjoob, M. and Yazdanpanah, R. (2013) 'Controlling of quadrotor uav using a fuzzy system for tuning the PID gains in hovering mode', in *10th Int. the International Conference on Advances in Computer Entertainment Technology*, pp.1–6.

Abdolhosseini, M., Zhang, Y.M. and Rabbath, C.A. (2013) 'An efficient model predictive control scheme for an unmanned quadrotor helicopter', *Journal of Intelligent & Robotic Systems*, Vol. 70, No. 1, pp.27–38.

Ahmed, N. and Chen, M. (2018) 'Sliding mode control for quadrotor with disturbance observer', *Advances in Mechanical Engineering*, Vol. 10, No. 7, p.1687814018782330.

Bangura, M. and Mahony, R. (2014) 'Real-time model predictive control for quadrotors', *IFAC Proceedings Volumes*, Vol. 47, No. 3, pp.11773–11780.

Borase, R.P., Maghade, D.K., Sondkar, S.Y. and Pawar, S.N. (2021) 'A review of PID control, tuning methods and applications', *International Journal of Dynamics and Control*, Vol. 9, No. 2, pp.818–827.

Boudjedir, H., Yacef, F., Bouhali, O. and Rizoug, N. (2012) 'Adaptive neural network for a quadrotor unmanned aerial vehicle', *International Journal in Foundations of Computer Science and Technology*, Vol. 2, No. 4, pp.1–13.

Bouhabza, N., Kara, K. and Hadjili, M.L. (2021) 'PID controllers design for a quadrotor system using teaching learning based optimization', *WSEAS Transactions on Applied and Theoretical Mechanics*, Vol. 16, pp.94–109, DOI: 10.37394/232011.2021.16.10.

Can, M.S. and Ercan, H. (2022) 'Real-time tuning of PID controller based on optimization algorithms for a quadrotor', *Aircraft Engineering and Aerospace Technology*, Vol. 94, No. 3, pp.418–430, https://doi.org/10.1108/AEAT-06-2021-0173.

Erginer, B. and Altuğ, E. (2012) 'Design and implementation of a hybrid fuzzy logic controller for a quadrotor VTOL vehicle', *International Journal of Control, Automation and Systems*, Vol. 10, No. 1, pp.61–70.

Ginting, A.H., Doo, S.Y., Pollo, D.E., Djahi, H.J. and Mauboy, E.R. (2022) 'Attitude control of a quadrotor with fuzzy logic controller on SO (3)', *Journal of Robotics and Control (JRC)*, Vol. 3, No. 1, pp.101–106.

Hassani, H., Mansouri, A. and Ahaitouf, A. (2022) 'Robust hybrid controller for quadrotor UAV under disturbances', *International Journal of Modelling, Identification and Control*, Vol. 40, No. 3, pp.195–203.

Hasseni, S.E.I., Abdou, L. and Glida, H.E. (2021) 'Parameters tuning of a quadrotor PID controllers by using nature-inspired algorithms', *Evolutionary Intelligence*, Vol. 14, No. 1, pp.61–73.

Ibraheem, I.K. and Ibraheem, G.A. (2016) 'Motion control of an autonomous mobile robot using modified particle swarm optimization based fractional order PID controller', *Eng. Technol. J.*, Vol. 34, No. 13, pp.2406–2419.

Jabeur, C.B. and Seddik, H. (2022) 'Optimized neural networks-PID controller with wind rejection strategy for a quad-rotor', *Journal of Robotics and Control (JRC)*, Vol. 3, No. 1, pp.62–72.

Joseph, S.B., Dada, E.G., Abidemi, A., Oyewola, D.O. and Khammas, B. (2022) 'Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems', *Heliyon*, Vol. 8, No 5, p.e09399.

Khuwaja, K., Tarca, I.C. and Tarca, R.C. (2018) 'PID controller tuning optimization with genetic algorithms for a quadcopter', *Recent Innovations in Mechatronics*, Vol. 5, No. 1, pp.1–7.

Maqsood, H. and Qu, Y. (2020) 'Nonlinear disturbance observer based sliding mode control of quadrotor helicopter', *Journal of Electrical Engineering & Technology*, Vol. 15, No. 3, pp.1453–1461.

Mofid, O., Mobayen, S. and Wong, W.K. (2020) 'Adaptive terminal sliding mode control for attitude and position tracking control of quadrotor UAVs in the existence of external disturbance', *IEEE Access*, Vol. 9, pp.3428–3440, DOI: 10.1109/ACCESS.2020.3047659.

Moshayedi, A.J., Gheibollahi, M. and Liao, L. (2020) 'The quadrotor dynamic modeling and study of meta-heuristic algorithms performance on optimization of PID controller index to control angles and tracking the route', *IAES International Journal of Robotics and Automation*, Vol. 9, No. 4, p.256.

Muhsen, A.N. and Raafat, S.M. (2021) 'Optimized PID control of quadrotor system using extremum seeking algorithm', *Engineering and Technology Journal*, Vol. 39, No. 6, pp.996–1010.

Muliadi, J. and Kusumoputro, B. (2018) 'Neural network control system of UAV altitude dynamics and its comparison with the PID control system', *Journal of Advanced Transportation*, Vol. 2018, Article ID 3823201, 18pp, https://doi.org/10.1155/2018/3823201.

Nadda, S. and Swarup, A. (2018) 'On adaptive sliding mode control for improved quadrotor tracking', *Journal of Vibration and Control*, Vol. 24, No. 14, pp.3219–3230.

Najm, A.A. and Ibraheem, I.K. (2019) 'Nonlinear PID controller design for a 6-DOF UAV quadrotor system', *Engineering Science and Technology, an International Journal*, Vol. 22, No. 4, pp.1087–1097.

Rao, R. and Patel, V. (2012) 'An elitist teaching-learning-based optimization algorithm for solving complex constrained optimization problems', *International Journal of Industrial Engineering Computations*, Vol. 3, No. 4, pp.535–560.

Rao, R.V., Savsani, V.J. and Vakharia, D.P. (2011) 'Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems', *Computer-Aided Design*, Vol. 43, No. 3, pp.303–315.

Rao, R.V., Savsani, V.J. and Vakharia, D.P. (2012) 'Teaching-learning-based optimization: an optimization method for continuous non-linear large scale problems', *Information Sciences*, Vol. 183, No. 1, pp.1–15.

Rodríguez-Abreo, O., Garcia-Guendulain, J.M., Hernández-Alvarado, R., Flores Rangel, A. and Fuentes-Silva, C. (2020) 'Genetic algorithm-based tuning of backstepping controller for a quadrotor-type unmanned aerial vehicle', *Electronics*, Vol. 9, No. 10, p.1735.

Saribas, H. and Kahvecioglu, S. (2021) 'PSO and GA tuned conventional and fractional order PID controllers for quadrotor control', *Aircraft Engineering and Aerospace Technology*, Vol. 93, No. 7, pp.1243–1253.

Seraji, H. (1998) 'A new class of nonlinear PID controllers with robotic applications', *Journal of Robotic Systems*, Vol. 15, No. 3, pp.161–181.

Siti, I., Mjahed, M., Ayad, H. and El Kari, A. (2019) 'New trajectory tracking approach for a quadcopter using genetic algorithm and reference model methods', *Applied Sciences*, Vol. 9, No. 9, p.1780.

Yu, G., Cabecinhas, D., Cunha, R. and Silvestre, C. (2019) 'Nonlinear backstepping control of a quadrotor-slung load system', *IEEE/ASME Transactions on Mechatronics*, Vol. 24, No. 5, pp.2304–2315.

Zeghlache, S., Saigaa, D., Harrag, A., Kara, K. and Bouguerra, A. (2012) 'Backstepping sliding mode controller improved with fuzzy logic: application to the quadrotor helicopter', *Archives of Control Sciences*, Vol. 22, No. 3, pp.315–342.