

Scientific-Research Article

Different Intelligent Methods for Coefficient Tuning of Quadrotor Feedback-linearization Controller

Mana Ghanifar¹, Milad Kamzan², Morteza Tayefi^{3*}

1- 2-3- Department of Aerospace Engineering, Khaje Nasir Toosi University of Technology

ABSTRACT

Keywords: Feedback-linearization, GA algorithm, PSO algorithm, Fuzzy-logic, Neural-network, Quadrotor UAV

This paper investigates different intelligent methods of tuning feedback-linearization control coefficients. Feedback-linearization technique is an effective method of controlling nonlinear systems. The most critical part of designing this controller is tuning the gains, especially if the plant has complex nonlinear dynamics. In this research, to improve the performance of the overall closed-loop system, the feedback linearization method has been integrated with the conventional proportional-integral-derivative (PID) controller. Also, a quadratic performance index was used to compare the functionality of the controllers tuned by the proposed intelligent methods. These intelligent methods include Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Fuzzy Logic, and Neural Network tuning algorithms. A quadrotor aircraft is used as the plant under study in order to evaluate the performance of the controllers tuned in this research. Finally, MATLAB simulation tests demonstrate the effectiveness of the presented algorithms.

Introduction

Unmanned Aerial Vehicles (UAVs) which are capable of remote and autonomous flying, have been continually and rapidly improving over the last few years. Generally, UAVs are divided into three categories: fixed-wing, rotary-wing, and flapping-wing aircrafts. In [1], H. Shraim et al. have introduced a characteristic table for each type. A classification of unmanned aerial vehicles was proposed by [2], which states that quadrotors are among the most well-known rotary-wing MAVs. Quadrotors, also known as quadcopters, are aircraft with four fixed pitch rotors. They belong to the heavier-than-air and rotary-wing categories of

aircraft, as shown in “Fig. 1” and “Fig. 2”. Quadrotors can move by using thrust and momentum generated by a difference in the angular velocity of their rotors. As for benefits, quadrotors have been used for load transportation [3]–[5], surveillance [6], mapping and taking aerial photos [7], [8], weather forecasting, and collecting data [9]. An overview of the mechanism of quadrotor flight has been provided by Shweta Gupte in [10]. It is challenging to control quadrotors because of the nonlinear under-actuated nature, strong coupling among flight modes, uncertainty, and nonlinear dynamic behavior of rotors. In spite of their high sensitivity to disturbances, quadrotors have advantages over fixed-wing aircraft, such as high

¹ PhD. Candidate

² PhD. Candidate

³ Assistant Professor ,(Correspondence Author) Email: Tayefi@kntu.ac.ir

manoeuvrability, vibration-free performance vertical take-off and landing, hovering, appropriate flight endurance, and a simple mechanical structure.

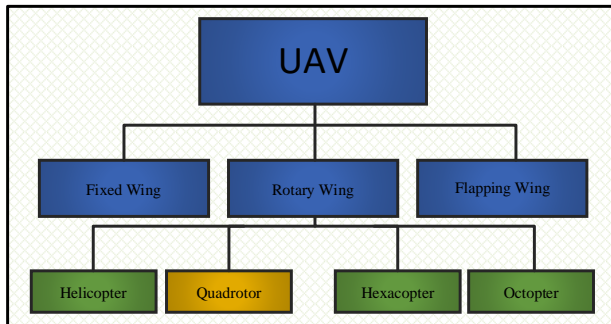


Fig. 1: UAVs classification [1].

One or more fast and high-performance controllers are required to eliminate the quadrotor's unstable nature. There have been many research projects on quadrotor control algorithms. Several control approaches, including backstepping and feedback-linearization, have been successfully applied and proven to function appropriately with quadrotors. [11] has comprehensively focused on quadrotor control. In the mentioned research, control methodologies are divided into three categories: linear such as PID, nonlinear such as feedback-linearization, and learning-based such as artificial neural network-based control algorithms. Also [12], [13] have reviewed some linear, and nonlinear control algorithms of quadrotors. Feedback-linearization is a common approach utilized in the control of nonlinear systems. This technique includes transforming the system into its equivalent linear form by producing appropriate control inputs. [14] has used this technique for quadrotors, though [15] has compared two control methods, including feedback-linearization and adaptive sliding mode control for a quadrotor helicopter. [16] has designed a more efficient controller, a combination of feedback-linearization and LQR.

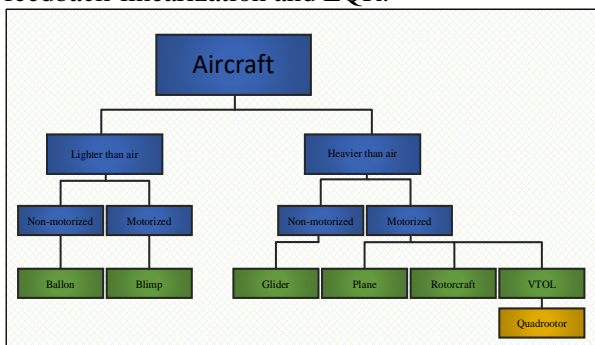


Fig. 2: Aircraft classification depending on flying principle and propulsion mode [10].

[17] has presented optimal controller based on feedback-linearization and linear quadratic regulator approaches using PSO to control a highly nonlinear quadrotor system. [18] has proposed a controller that combines feedback-linearization and Embedded Model Control (EMC) framework. Also, an innovative control architecture for quadrotors is proposed in [19] which utilizes feedback-linearization method. In this research, with the aim of improving the total behaviour and increasing the robustness of the controlled system, a combination of the feedback linearization method and the proportional-integral-derivative (PID) controller has been used.

The main contributions of this paper are summarized as follows: First, four different methods for calculating proper controller coefficients are discussed and implemented. Second, a comparison between the results from each technique is made, which shows that the two online methodologies lead to a better CLS performance. *The remainder of this paper* is structured as follows: First, mathematical model of a quadrotor UAV is presented in the following section. In the next section, feedback-linearization formulations are provided along with this technique's application in controlling the system. In the third step, four different methods are employed to determine the appropriate control coefficients. Lastly, the performances of the aforementioned techniques are evaluated based on their relative cost functions.

Dynamic Model

As previously mentioned, the quadrotor is a nonlinear dynamic system with four fixed pitch angle rotors that provides the necessary forces and moments to flight. These rotors are driven by four brushless DC motors with high-speed response, allowing the system to execute commands rapidly. "Fig. 3" illustrates the proposed quadrotor, showing the forces and moments on the rotors.

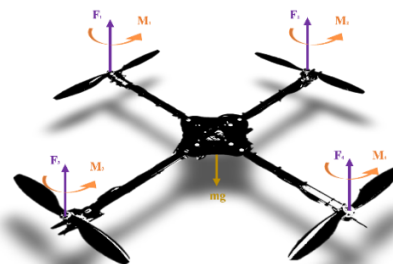


Fig. 3. Forces and moments acting on a quadrotor UAV.

To provide a mathematical model for the quadrotor, the model introduced in [18] is employed. In this derivation of system dynamics, the quadrotor is modeled by six second-order nonlinear differential equations. The main advantage of utilizing these equations in the control process is to reduce the number of necessary controllers. This model is presented in (1) to (6):

$$\ddot{X} = (\cos \varphi \cos \psi \sin \theta + \sin \varphi \sin \psi) \frac{U_1}{m} \quad (1)$$

$$\ddot{Y} = (\cos \varphi \sin \psi \sin \theta - \cos \psi \sin \varphi) \frac{U_1}{m} \quad (2)$$

$$\ddot{Z} = -g + (\cos \theta \cos \varphi) \frac{U_1}{m} \quad (3)$$

$$\ddot{\varphi} = \frac{u_2}{I_{xx}} \quad (4)$$

$$\ddot{\theta} = \frac{u_3}{I_{yy}} \quad (5)$$

$$\ddot{\psi} = \frac{u_4}{I_{zz}} \quad (6)$$

Where X, Y and Z refer to system longitudinal, lateral and vertical position and φ , θ , and ψ show the system Euler angles. Furthermore, the constant parameters m, g, and I correspond to system total mass, gravitational acceleration, and inertia tensor components.

In a 2D vertical plane, the system dynamics can be presented as a 3DOF mathematical model. This model is derived from the previous equations with the $X = \theta \approx 0$ approximation. “Fig. 4” shows the system configuration in this situation.

$$\ddot{Y} = -\frac{u_1}{m} \sin \varphi \quad (7)$$

$$\ddot{Z} = -g + \frac{u_1}{m} \cos \varphi \quad (8)$$

$$\ddot{\varphi} = \frac{u_2}{I_{xx}} \quad (9)$$

In order to analyze and investigate the performance of the proposed controller, we consider a sinusoidal disturbance, shown in “Fig. 4”, with a constant amplitude and frequency [20].

$$D = \sin(3t) \quad (10)$$

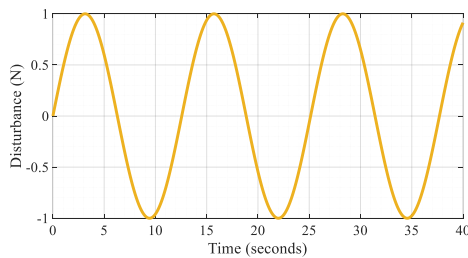


Fig. 4 Input disturbance.

Controlling the system based on the feedback-linearization method

In this paper, feedback-linearization method is utilized to control the nonlinear dynamic model of the quadrotor system. Using this technique, results in removing as much nonlinearity as possible from the dynamic model of the system which leads to a semi-linear behaviour of the closed-loop system. “Fig. 5” illustrates the proposed quadrotor, showing the forces and moments on the rotors. The equations of control input based on FL-PID¹ are as follows:

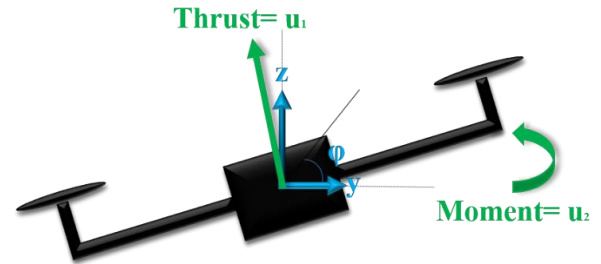


Fig. 5. Quadrotor in 2D space

$$u_1 = \frac{m}{\cos \varphi} (g + \ddot{z}_{des} + k_{d,z}(\dot{z}_{des} - \dot{z}) + k_{p,z}(z_{des} - z) + k_{i,z} \left(\int z_{des} - \int z \right) + D) \quad (11)$$

$$u_2 = I_{xx} (\ddot{\varphi}_c + k_{p,\varphi}(\varphi_c - \varphi) + k_{d,\varphi}(\dot{\varphi}_c - \dot{\varphi}) + k_{i,\varphi} \left(\int \varphi_c - \int \varphi \right) + D) \quad (12)$$

Where z_{des} , \dot{z}_{des} and \ddot{z}_{des} refer to desired vertical position, velocity and acceleration, respectively. Also φ_c , $\dot{\varphi}_c$ and $\ddot{\varphi}_c$ are the command roll angles generated by the position controller and its first and second derivatives. As a final note, k_p , k_d and k_i indicate the coefficients of the controllers, each separated by the channel subscript.

Different Tuning Methods

Four different approaches are used to adjust the closed-loop quadrotor system's control parameters. These methods can be classified into two categories: online and offline coefficient tuning algorithms. The mentioned offline methods, include genetic algorithm and particle swarm optimization, while online approaches include neural networks and fuzzy logic.

¹ feedback linearization- proportional integral derivative

Genetic Algorithm (GA) and Particle swarm optimization (PSO)

First, the closed-loop controller coefficients are calculated by using two offline intelligent optimization algorithms [21], [22]. These algorithms are the genetic algorithm and particle swarm. In practice, even though the computational base of both techniques is similar in some ways, they are the result of two independent algorithms [23]. They both begin with an initial population, then advance based on their logic, inspired by their relative natural phenomena. Finally, these algorithms are able to minimize the value of the predefined optimization index (the cost function) which in this paper is defined in a quadratic form as follows:

$$J = \mathbf{x}^T(t_f)\mathbf{E}\mathbf{x}(t_f) + \int_{t_0}^{t_f} (\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}) dt \quad (13)$$

Here J is the cost function value, \mathbf{R} , \mathbf{Q} and \mathbf{E} are the weighting matrices, respectively. Also, the variables \mathbf{x} and \mathbf{u} are the system state and control input vectors respectively. The controller parameters which are adjusted by each of the offline optimization algorithms are presented in “Table 1”, and “Table 2”. Also, “Fig. 6” and “Fig. 7” illustrate the changes in the value of the performance index (J) as a function of iterations during the optimization process with each of the mentioned algorithms. Finally, “Fig. 8”, demonstrates the changes in the system's ϕ -orientation during the simulation as a function of time.

Table 1 Calculated coefficients using PSO.

	Kp	Ki	Kd
z	2.6543	0.0112	3.0000
y	1.0100	0.0100	1.5587
ϕ	660.000	0.0400	125.0000

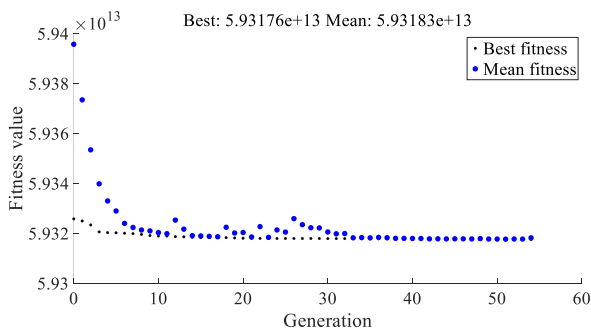


Fig. 6. Fitness value of GA optimization as a function of generation.

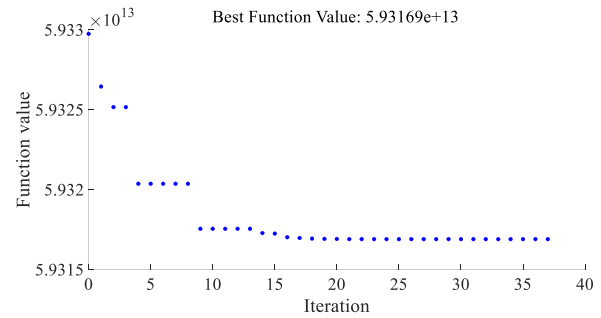


Fig. 7. Fitness value of PSO optimization as a function of iteration.

Table 2 Calculated coefficients using GA.

	Kp	Ki	Kd
z	2.6553	0.0112	3.0003
y	1.0099	0.0111	1.5575
ϕ	659.9957	0.0469	125.0210

The overall algorithm of control coefficient tuning using GA and PSO is represented as a code box in Fig. 9.

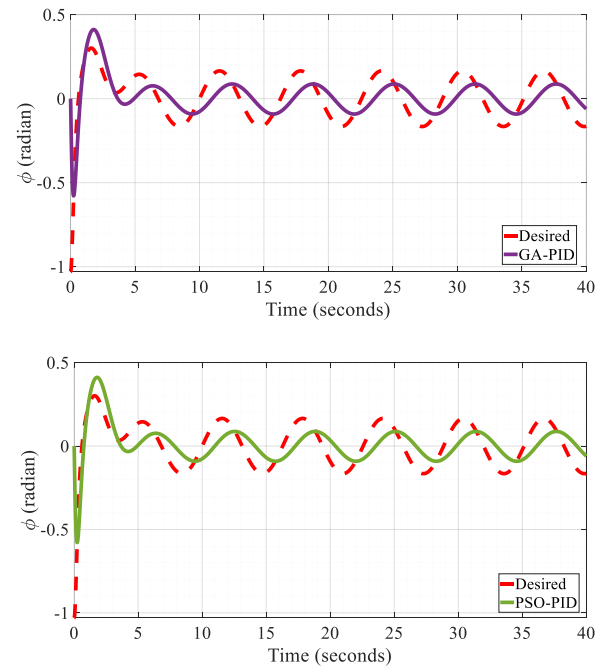


Fig. 8. ϕ -orientation as a function of time
a)GA-PID and b) PSO-PID.

1. **Initialization**
2. $m \leftarrow \text{Mass} = 0.52 \text{ kg}$
3. $I \leftarrow \text{Moment of inertia} = 0.006228 \text{ kg.m}^2$
4. $g \leftarrow \text{standard gravity} = 9.8 \text{ m/s}^2$
5. $f \leftarrow \text{Frequency of Sinusoidal Disturbance} = 3 \text{ Hz}$
6. $d \leftarrow \text{Amplitude of Sinusoidal Disturbance} = 1$
7. $t \leftarrow \text{Time}$
8. $dt \leftarrow \text{Sample Time} = 0.01 \text{ s}$
9. $\text{lb} \leftarrow \text{Lower Bounds} = [2, 0.01, 3, 640, 0.04, 125, 1, 0.01, 1]$
10. $\text{ub} \leftarrow \text{Upper Bounds} = [3, 0.02, 3.01, 660, 0.06, 135, 1.01, 0.02, 2]$
11. $\mathbf{k} \leftarrow \text{FL - PID coefficients}, \text{lb} \leq \mathbf{k} \leq \text{ub}$
12. **Time Span** $\leftarrow 0:dt:40$
13. $D \leftarrow \text{Disturbance} = d * \sin(f * t)$
14. $T \leftarrow \text{Target Value} = 10 \text{ m}$
15. $\text{Method} \leftarrow \text{GA or PSO}$
16. \mathbf{H}, \mathbf{Q} and $\mathbf{R} \leftarrow \text{weighting matrices}$
17. $J = \mathbf{x}_{tf}' * \mathbf{H} * \mathbf{x}_{tf} + \int_0^{tf} (\mathbf{x}' * \mathbf{Q} * \mathbf{x} + \mathbf{u}' * \mathbf{R} * \mathbf{u})$
18. **if** PSO
 - o $\text{SwarmSize} \leftarrow 100$
 - o $\text{MaxIterations} \leftarrow 1800$
 - o $\text{MaxTime} \leftarrow \text{Inf}$
 - o $\text{Tolerance} \leftarrow 10^{-6}$
 - o **Nonlinear Mathematical Model**
19. **elseif** GA
 - o $\text{PopulationSize} \leftarrow 100$
 - o $\text{MaxGenerations} \leftarrow 100$
 - o $\text{MaxTime} \leftarrow \text{Inf}$
 - o $\text{Tolerance} \leftarrow 10^{(-6)}$
 - o **Nonlinear Mathematical Model**
20. **end if**

Fig. 9. Gain tuning algorithm based on GA/PSO.

Fuzzy Logic

A mamdani fuzzy system is utilized as another technique in this study to determine the appropriate control coefficients. Defined fuzzy rules (adapted from [24]–[27]) choose the proper coefficient and update these coefficients in an online way based on the value and derivative of the error at any given sampling time, within the set range for each of them. The Block diagram of the fuzzy tuning closed loop system is shown in Fig. 10. Using the predefined cost function (J), the value of this index is determined at the end of the process. The fuzzy meshes in the z-channel control are shown in “Fig. 11”. Following is a code box

representing the algorithm of fuzzy tuning the control coefficients in this method, “Fig. 12”.

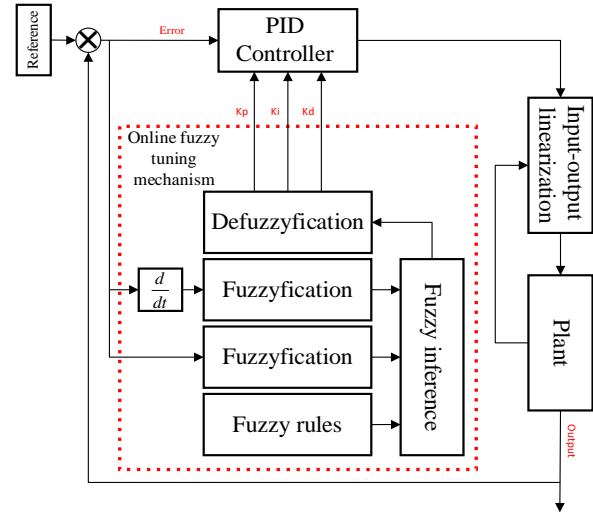


Fig. 10. Block diagram of closed loop system with fuzzy tuning mechanism.

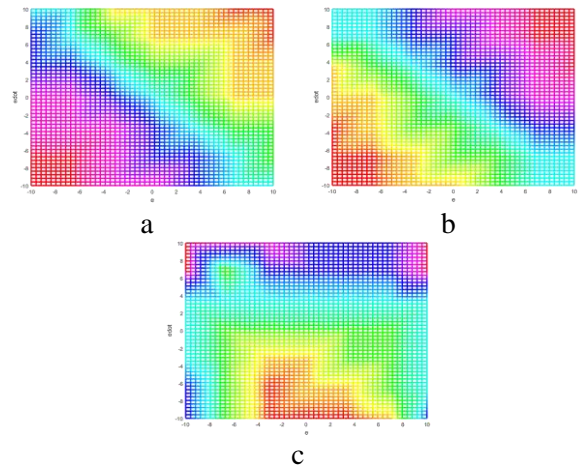


Fig. 11. Hsv-mesh of fuzzy rules[28], a) Kp, b) Ki, c) Kd

1. **Initialization**
2. $m \leftarrow \text{Mass} = 0.52 \text{ kg}$
3. $I \leftarrow \text{Moment of inertia} = 0.006228 \text{ kg.m}^2$
4. $g \leftarrow \text{standard gravity} = 9.8 \text{ m/s}^2$
5. $f \leftarrow \text{Frequency of Sinusoidal Disturbance} = 3 \text{ Hz}$
6. $d \leftarrow \text{Amplitude of Sinusoidal Disturbance} = 1$
7. $t \leftarrow \text{Time}$
8. $dt \leftarrow \text{Sample Time} = 0.01 \text{ s}$
9. $\text{lb} \leftarrow \text{Lower Bounds} = [2, 0.01, 3, 640, 0.04, 125, 1, 0.01, 1]$
10. $\text{ub} \leftarrow \text{Upper Bounds} = [3, 0.02, 3.01, 660, 0.06, 135, 1.01, 0.02, 2]$
11. **Time Span** $\leftarrow 0: dt: 40$
12. $p \leftarrow \text{Number of Sample Time}$
13. $D \leftarrow \text{Disturbance} = d * \sin(f * t)$
14. $T \leftarrow \text{Target Value} = 10 \text{ m}$
15. **Method** $\leftarrow \text{Fuzzy Logic}$
16. **for** $i = 1:1:p$
 - o $e = T - x(j), j = 1, 3 \text{ or } 5$
 - o **Fuzzy Rules** $\leftarrow \text{input: } [e, T, x(j)], \text{output: } \mathbf{k}$
 - o $\mathbf{k} \leftarrow \text{FL - PID coefficients, } \text{lb} \leq \mathbf{k} \leq \text{ub}$
 - o **Nonlinear Mathematical Model**
17. **end for**
18. $\mathbf{H}, \mathbf{Q} \text{ and } \mathbf{R} \leftarrow \text{weighting matrices}$
19. $J = \mathbf{x}_{tf}' * \mathbf{H} * \mathbf{x}_{tf} + \int_0^{tf} (\mathbf{x}' * \mathbf{Q} * \mathbf{x} + \mathbf{u}' * \mathbf{R} * \mathbf{u})$

Fig. 12. Gain tuning algorithm based on fuzzy logic.

Neural Network

To achieve the main goal of this research, a two-layered dynamic neural network is used. The error, desired value, and actual value of states are network's inputs, while the control coefficients of the controllers are considered as the mechanism outputs. Note that, the activation functions employed in the layers are tansig. The Block diagram of the neural tuning closed loop system is shown in Fig. 13. In order to obtain appropriate results, it is essential to initialize the network weights and biases precisely. Initial weights of the neural network in this paper are adjusted properly through trial and error to ensure that responses are obtained within the minimum possible time and with the lowest possible performance index value. Also, Fig.16 demonstrates the code box for tuning the control gains using this method.

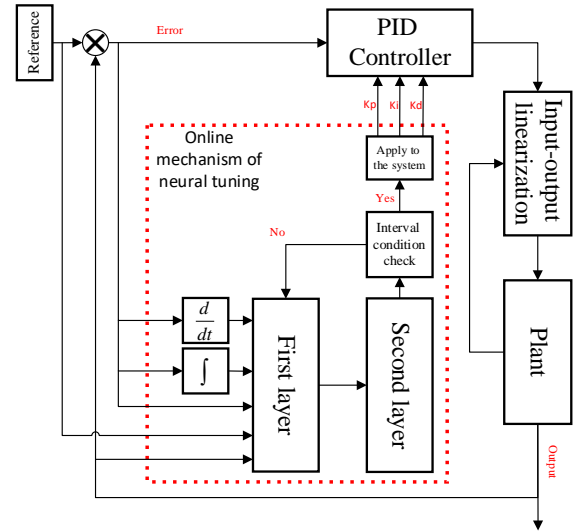


Fig. 13. Block diagram of closed loop system with Neural tuning mechanism.

1. **Initialization**
2. $m \leftarrow \text{Mass} = 0.52 \text{ kg}$
3. $I \leftarrow \text{Moment of inertia} = 0.006228 \text{ kg.m}^2$
4. $g \leftarrow \text{standard gravity} = 9.8 \text{ m/s}^2$
5. $f \leftarrow \text{Frequency of Sinusoidal Disturbance} = 3 \text{ Hz}$
6. $d \leftarrow \text{Amplitude of Sinusoidal Disturbance} = 1$
7. $t \leftarrow \text{Time}$
8. $dt \leftarrow \text{Sample Time} = 0.01 \text{ s}$
9. $\text{lb} \leftarrow \text{Lower Bounds} = [2, 0.01, 3, 640, 0.04, 125, 1, 0.01, 1]$
10. $\text{ub} \leftarrow \text{Upper Bounds} = [3, 0.02, 3.01, 660, 0.06, 135, 1.01, 0.02, 2]$
11. **Time Span** $\leftarrow 0: dt: 40$
12. $p \leftarrow \text{Number of Sample Time}$
13. $D \leftarrow \text{Disturbance} = d * \sin(f * t)$
14. $T \leftarrow \text{Target Value} = 10 \text{ m}$
15. **Method** $\leftarrow \text{Neural Network (1000 iterations)}$
16. **for** $i = 1:1:p$
 - o $e = T - x(j), j = 1, 3 \text{ or } 5$
 - o **2 - layer MLP** $\leftarrow \text{input: } [e, T, x(j)], \text{output: } \mathbf{k}$
 - o $\mathbf{k} \leftarrow \text{FL - PID coefficients, } \text{lb} \leq \mathbf{k} \leq \text{ub}$
 - o **Nonlinear Mathematical Model**
17. **end for**
18. $\mathbf{H}, \mathbf{Q} \text{ and } \mathbf{R} \leftarrow \text{weighting matrices}$
19. $J = \mathbf{x}_{tf}' * \mathbf{H} * \mathbf{x}_{tf} + \int_0^{tf} (\mathbf{x}' * \mathbf{Q} * \mathbf{x} + \mathbf{u}' * \mathbf{R} * \mathbf{u})$

Fig. 14. Gain tuning algorithm based on neural network.

Comparison of Simulation Results

From the comparison of the results obtained from the simulation of the behaviour of the closed-loop system, it is inferred that online controller coefficient adjustment techniques generally provide more accurate results and better performance for closed-loop systems. It is mainly because they can rapidly change control coefficients in response to system states, error values (including actual values, derivatives, and integrals of error), disturbances, and reference inputs (setpoints or reference paths). As previously mentioned, online tuning procedures' benefits can ultimately improve closed-loop properties. Using these online or offline techniques, "Table 3" shows the value of the cost function in each case.

Table 3. The value of the cost function with different coefficient adjustment techniques.

Tuning method	Cost function value
Genetic Algorithm	$5.98932670 \times 10^{13}$
Particle Swarm	$5.98919352 \times 10^{13}$
Fuzzy Logic	$5.94722569 \times 10^{13}$
Neural Network	$5.85966716 \times 10^{13}$

"Fig. 15", "Fig. 16" and "Fig. 17" show the CLTF2 response of the proposed quadrotor system. The control parameters are obtained using the neural network adjustment algorithm. It is crucial that the control efforts are kept within a reasonable range in this control process. "Fig. 18" and "Fig. 19" show the changes in the control inputs.

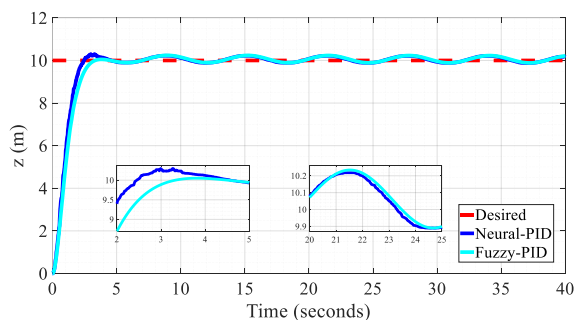


Fig. 15. z-position as a function of time(Neural-PID, Fuzzy-PID.)

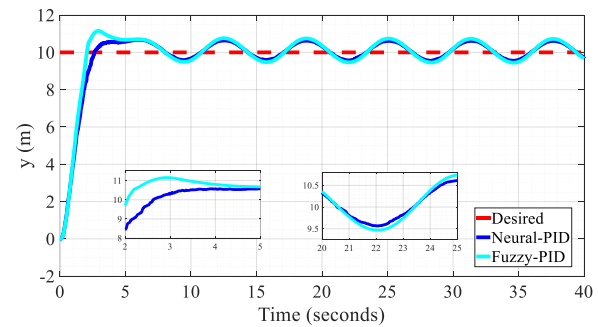


Fig. 16. y-position a function of time (Neural-PID, Fuzzy-PID)

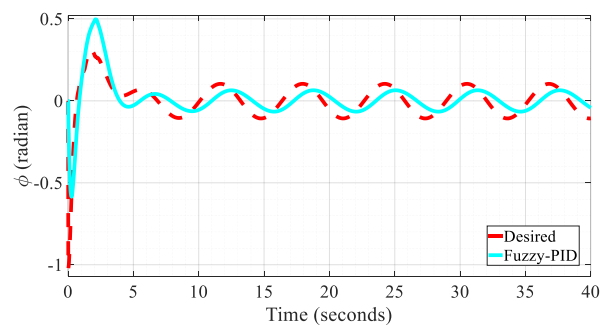
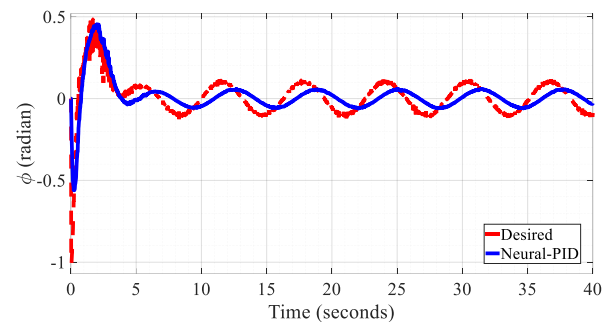


Fig. 17. ϕ -orientation as a function of time
a)Neural-PID and b)Fuzzy-PID

For the purpose of limiting the volume of the article, only plots related to the PID coefficients of z position are presented in this section ("Fig. 20", "Fig. 21" and "Fig. 22").

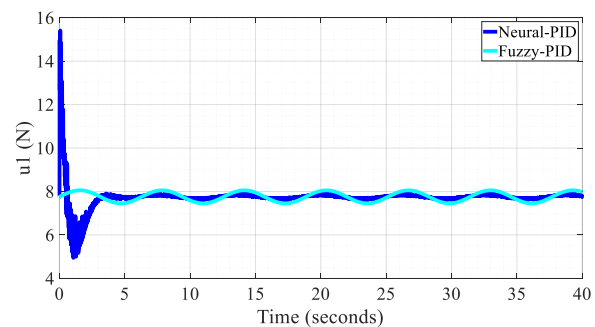


Fig. 18. u_1 -input a function of time (Neural-PID, Fuzzy-PID)

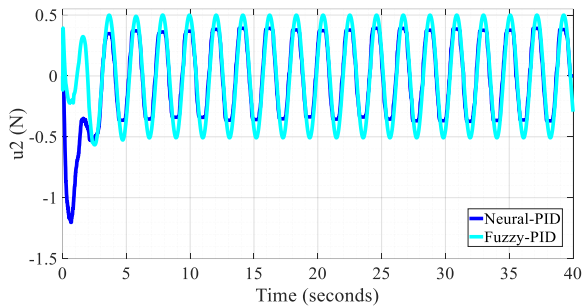


Fig. 19. u_2 -input a function of time (Neural-PID, Fuzzy-PID)

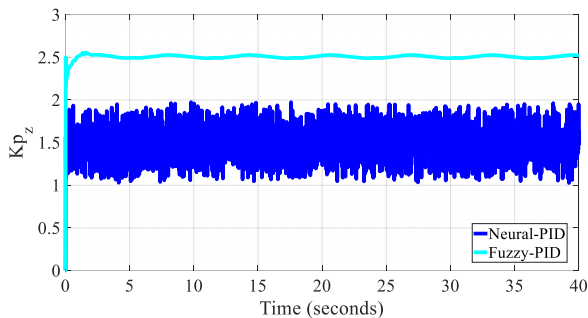


Fig. 20. K_{p_z} -coefficient a function of time (Neural-PID, Fuzzy-PID)

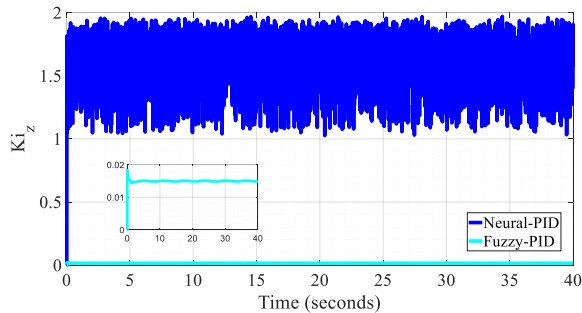


Fig. 21. K_{i_z} -coefficient a function of time (Neural-PID, Fuzzy-PID)

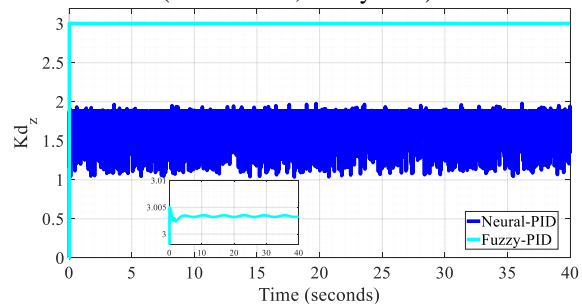


Fig. 22. K_{d_z} -coefficient a function of time (Neural-PID, Fuzzy-PID)

Conclusion

This paper examines four intelligent coefficient tuning approaches to achieve FL-PID control gains in the quadrotor flight simulation. These four strategies are divided into online (neural network algorithms and fuzzy logic) and offline (genetic

algorithms and particle swarm optimization). By comparing the final values of the predefined cost function generated in each case based on the outputs in case of disturbance, it was shown that the calculation of control coefficients using online techniques can significantly reduce the cost function. The main reason for this event is the high compatibility of online methods in the whole simulation process.] In other words, these methods constantly try to make the response of the system as close to the target as possible by determining each of the control coefficients accurately.

References

- [1] H. Shraim, A. Awada, and R. Youness, "A survey on quadrotors: Configurations, modeling and identification, control, collision avoidance, fault diagnosis and tolerant control," *IEEE Aerospace and Electronic Systems Magazine*, vol. 33, no. 7, pp. 14–33, 2018, doi: 10.1109/MAES.2018.160246.
- [2] M. Hassanalain and A. Abdelkefi, "Classifications, applications, and design challenges of drones: A review," *Progress in Aerospace Sciences*, vol. 91, May 2017, doi: 10.1016/j.paerosci.2017.04.003.
- [3] I. H. B. Pizetta, A. S. Brandao, and M. Sarcinelli-Filho, "Load Transportation by Quadrotors in Crowded Workspaces," *IEEE Access*, vol. 8, pp. 223941–223951, 2020, doi: 10.1109/ACCESS.2020.3043719.
- [4] D. K. D. Villa, A. S. Brandão, and M. Sarcinelli-Filho, "A Survey on Load Transportation Using Multirotor UAVs," *Journal of Intelligent & Robotic Systems* 2019 98:2, vol. 98, no. 2, pp. 267–296, Oct. 2019, doi: 10.1007/S10846-019-01088-W.
- [5] G. Ononiwu, O. Onojo, O. Ozioko, and O. Nosiri, "Quadcopter Design for Payload Delivery," *Journal of Computer and Communications*, vol. 04, no. 10, pp. 1–12, 2016, doi: 10.4236/jcc.2016.410001.
- [6] P. N. Patel, M. Patel, R. Faldu, and Y. R. Dave, "Quadcopter for Agricultural Surveillance," *Advance in Electronic and Electric Engineering*, pp. 427–432, 2013.
- [7] W. Budiharto, E. Irwansyah, J. S. Suroso, A. Chowanda, H. Ngarianto, and A. A. S. Gunawan, "Mapping and 3D modelling using quadrotor drone and GIS software," *J Big Data*, vol. 8, no. 1, p. 48, Dec. 2021, doi: 10.1186/s40537-021-00436-8.
- [8] B. Chamberlain and W. Sheikh, "Design and Implementation of a Quadcopter Drone Control System for Photography Applications," in *2022 Intermountain Engineering, Technology and Computing (IETC)*, May 2022, pp. 1–7. doi: 10.1109/IETC54973.2022.9796735.
- [9] S. Kefei, L. Baoying, L. Hanxu, and L. Chen, "Agricultural Environment Monitoring Combined with Quadrotor Aircraft Control Algorithm," *Engineering Science and Technology Review*, pp. 190–200, 2019.
- [10] S. Gupte, P. I. T. Mohandas, and J. M. Conrad, "A survey of quadrotor unmanned aerial vehicles," *Conference Proceedings - IEEE SOUTHEASTCON*, 2012, doi: 10.1109/SECon.2012.6196930.
- [11] R. Amin, L. Aijun, and S. Shamshirband, "A review of quadrotor UAV: control methodologies and performance evaluation," *International Journal of Automation and*

- Control*, vol. 10, no. 2, pp. 87–103, May 2016, doi: 10.1504/IJAAC.2016.076453.
- [12] J. Kim, S. A. Gadsden, and S. A. Wilkerson, “A Comprehensive Survey of Control Strategies for Autonomous Quadrotors,” *Canadian Journal of Electrical and Computer Engineering*, vol. 43, no. 1, pp. 3–16, 2020, doi: 10.1109/CJECE.2019.2920938.
- [13] Lebao Li, Lingling Sun, and Jie Jin, “Survey of advances in control algorithms of quadrotor unmanned aerial vehicle,” in *2015 IEEE 16th International Conference on Communication Technology (ICCT)*, Oct. 2015, pp. 107–111. doi: 10.1109/ICCT.2015.7399803.
- [14] B. Song, Y. Liu, and C. Fan, “Feedback linearization of the nonlinear model of a small-scale helicopter,” *J Control Theory Appl*, vol. 8, no. 3, pp. 301–308, 2010, doi: 10.1007/s11768-010-0017-8.
- [15] D. Lee, H. J. Kim, and S. Sastry, “Feedback Linearization vs. Adaptive Sliding Mode Control for a Quadrotor Helicopter,” *Int J Control Autom Syst*, vol. 7, no. 3, pp. 419–428, 2009, doi: 10.1007/s12555-009-0311-8.
- [16] Z. Shulong, A. Honglei, Z. Daibing, and S. Lincheng, “A new feedback linearization LQR control for attitude of quadrotor,” *2014 13th International Conference on Control Automation Robotics and Vision, ICARCV 2014*, pp. 1593–1597, 2014, doi: 10.1109/ICARCV.2014.7064553.
- [17] C.-C. Chen and Y.-T. Chen, “Feedback Linearized Optimal Control Design for Quadrotor with Multi-performances,” *IEEE Access*, pp. 1–1, Feb. 2021, doi: 10.1109/access.2021.3057378.
- [18] M. A. Lotufo, L. Colangelo, and C. Novara, “Feedback Linearization for Quadrotors UAV,” Jun. 2019, doi: <https://doi.org/10.48550/arXiv.1906.04263>.
- [19] L. Martins, C. Cardeira, and P. Oliveira, “Inner-outer feedback linearization for quadrotor control: two-step design and validation,” *Nonlinear Dynamics 2022 110:1*, vol. 110, no. 1, pp. 479–495, Jun. 2022, doi: 10.1007/S11071-022-07632-Y.
- [20] W. C. Gan, L. Qiu, and J. Wang, “An Adaptive Sinusoidal Disturbance Rejection Controller for Single-Input-Single-Output Systems,” *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 15684–15689, 2008, doi: 10.3182/20080706-5-KR-1001.02652.
- [21] N. el Gmili, M. Mjahed, A. el Kari, and H. Ayad, “Intelligent PSO-based PDs/PIDs controllers for an unmanned quadrotor,” *International Journal of Intelligent Engineering Informatics*, vol. 6, no. 6, p. 548, 2018, doi: 10.1504/IJIEI.2018.096579.
- [22] D. T. Pham and D. Karaboga, “Intelligent optimisation techniques : genetic algorithms, tabu search, simulated annealing and neural networks,” p. 302, 2000.
- [23] V. Kachitvichyanukul, “Comparison of Three Evolutionary Algorithms: GA, PSO, and DE,” *Industrial Engineering and Management Systems*, vol. 11, no. 3, pp. 215–223, Sep. 2012, doi: 10.7232/iems.2012.11.3.215.
- [24] Timothy J. Ross, *Fuzzy Logic with Engineering Applications*, Second., no. John Wiley & Sons. John Wiley & Sons, 2004.
- [25] A. Visioli, “Tuning of PID controllers with fuzzy logic,” *IEE Proceedings: Control Theory and Applications*, vol. 148, no. 1, pp. 1–8, Jan. 2001, doi: 10.1049/IP-CTA:20010232.
- [26] Seyed Mostafa Kia, *Fuzzy Logic in MATLAB*. Kian Publication.
- [27] Zhen-Yu Zhao, M. Tomizuka, and S. Isaka, “Fuzzy gain scheduling of PID controllers,” *IEEE Trans Syst Man Cybern*, vol. 23, no. 5, pp. 1392–1398, 1993, doi: 10.1109/21.260670.
- [28] Ghanifar Mana, Kamzan Milad, and Tayefi Morteza, “PID gain tuning using intelligent adaptive and non-adaptive algorithms: with implementation in a quadrotor ,” in *The 19th International Conference of Iranian Aerospace Society*, 2021. Accessed: Jan. 19, 2022. [Online]. Available: <https://civilica.com/doc/1362264/>

COPYRIGHTS

©2023 by the authors. Published by Iranian Aerospace Society This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 International (CC BY 4.0)
<https://creativecommons.org/licenses/by/4.0/>

