

Research Article

Neural Network Control System of UAV Altitude Dynamics and Its Comparison with the PID Control System

Jemie Muliadi and Benyamin Kusumoputro 

Department of Electrical Engineering, Universitas Indonesia, Kampus Baru UI, Depok 16424, Indonesia

Correspondence should be addressed to Benyamin Kusumoputro; kusumo@ee.ui.ac.id

Received 5 May 2017; Revised 25 July 2017; Accepted 16 August 2017; Published 22 January 2018

Academic Editor: Cheng S. Chin

Copyright © 2018 Jemie Muliadi and Benyamin Kusumoputro. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This article proposes a comparative method to assess the performance of artificial neural network's direct inverse control (DIC-ANN) with the PID control system. The comparison served as an analysis tool to assess the advantages of DIC-ANN over conventional control method for a UAV attitude controller. The development of ANN method for UAV control purposes arises due to the limitations of the conventional control method, which is the mathematical based model, involving complex expression, and most of them are difficult to be solved directly into analytic solution. Although the linearization simplified the solving process for such mathematical based model, omitting the nonlinear and the coupling terms is unsuitable for the dynamics of the multirotor vehicle. Thus, the DIC-ANN perform learning mechanism to overcome the limitation of PID tuning. Therefore, the proposed comparative method is developed to obtain conclusive results of DIC-ANN advantages over the linear method in UAV attitude control. Better achievement in the altitude dynamics was attained by the DIC-ANN compared to PID control method.

1. Introduction

The advancement of the Unmanned Aerial Vehicle (UAV) has triggered the development of the autonomous technology for various assignments, from the applied strategic and tactical mission to the simple carrier of the video shooting camera and, recently, the SAR purposes [1, 2]. Meanwhile, in the field of autonomous technology, UAVs already become the preferred platform for research and development of the flight control systems. Numerous types of UAV have been constructed and developed for research platform, and between those types, the quadrotor becomes the popular type. The popularity emerged because of the construction's simplicity, the easiness of maintenance, their ability to hover, and the capability of performing vertical take-off and landing (VTOL) [3]. The quadrotor also provides two advantages from the traditional rotorcraft vehicle like the helicopter [3]. First, the quadrotor does not require complicated linkages to actuate its rotor; instead, the flight was controlled by varying the angular speed of each rotor. Second, by using four propellers instead of one, the quadrotor required smaller diameter for propellers.

The quadrotor is a basically unstable aerial vehicle [4]. Such instability—although quadrotor has been successfully flying in 1920—has inhibited the development of the practical quadrotor until recent years. The instability occurred because of the difficulties of controlling all of the four rotors simultaneously with a proper amount of bandwidth [3]; until these days, due to the availability of sophisticated flight controller board, one can fly the unmanned quadrotor using the aeromodelling remote controller. These conventional flight controllers, including the commercially produced, were based on the PID control method and designed by mathematical based model.

The PID control method is a common method of control which processes the error signal into input signal by the aid of proportional factor (P), integrative action (I), and differential action (D). The states were compared with their respective references (the set points) to obtain the error signal. The error signal amplified with the proportional gain (K_p), interpreted as the present error term, to improve the speed of the system response to its error. In parallel, the error signal also integrated together with the integral gain (K_i) as the accumulation of the past error term and being

differentiated together with the derivative gain (K_D) as the prediction of future error term [5]. These gains were adjusted with the tuning methods based on the linear mathematical model that represents the dynamics of the system. The PID gains can be tuned easily for the systems with a low level of nonlinearity and perfectly uncoupled between the modes of its dynamics. On the contrary, the quadrotor flight dynamics is a highly nonlinear, strongly coupled, multivariable, and also underactuated system [4]. Hence, these contraries stimulated various research about quadrotor flight control methods besides PID.

Alongside the linear control method like PID, the nonlinear methods are also developed widely from the underwater robot like ROV [6, 7] and AUV [8, 9], to the aerial robot like quadrotor. Several applied methods for the quadrotor and unmanned robots are the backstepping control [10–13], sliding mode control [9, 14–17], and the learning-based control method like fuzzy logic and artificial neural-network-based controllers [18]. This article compared one of the artificial neural-network-based controllers, that is, the artificial neural network's direct inverse control (DIC-ANN), with the PID controller in the quadrotor attitude dynamics.

Artificial neural network (ANN) is a parallel distributed processor consisting of simple neurons that enabled memorizing the knowledge of a system by mimicking its nonlinear model after training process [19]. In the control areas, the ANN-based control system arises due to the drawbacks of the nonlinear control systems like backstepping, sliding mode controller, and so forth. The backstepping controller has a fast converge rate and can handle uncertainties, but the robustness is bad; while the sliding mode controller has good robustness and simple physical implementation, it suffers the chattering phenomena [20].

The ANN worked as regressor structure which minimizes an error function in an iterative process by applying the backpropagation of error algorithm. Since ANN is a non-parametric model consisting of connected neurons in several layers, the training process will adjust the weight in every neuron's connection to obtain the optimum fitting of system dynamics representation. For control purposes, the ANN has been used to invert the system dynamics to obtain the input signal correlated with the desired output results. The inversion technique initiates the direct inverse control (DIC) method with the use of ANN. The ANN method has been applied for nonlinear systems from chemical reactor [21] to nuclear purposes [22], whereas, in the field of unmanned system's controller, the DIC-ANN becomes research topics in UAV, for example, hexacopter [19], helicopter [23], and quadrotor [24] to the AUV [8].

This article proposed the method of comparing the DIC-ANN with the PID control method. The research of DIC-ANN in UAV controller [19, 23, 24] has published the advantages of the method for UAV application. However, to validate their improvement, the results still not be comparable with the conventional control method. Hence this article assesses the requirement of such comparison for the validation purpose. The method was derived to perform a "head to head" comparison to measure the improvement occurred.

To give insight into the proposed comparative method between ANN and PID control in quadrotor attitude, the paper will be organized as follows. After introductory narration in the first section, Section 2 will address the PID control system representing the dominant conventional control method for UAV. To introduce the ANN control system, Section 3 will present the DIC-ANN control system for quadrotor attitude dynamics. Since the ANN-based control appears to have advantages compared to the conventional control method, Section 4 will discuss the comparison method required for validation purposes. Continued from the previous sections, the comparative analysis will be performed in Section 5 to validate the advantages of DIC-ANN over PID control method. Finally, Section 6 will conclude the work in this article.

2. PID Control of Quadrotor Dynamics

The PID is a popular method of control for various system dynamics. Besides the wide application in the industry, the PID was also applied in UAV control systems. Including the quadrotor, PID also become a common method to build the flight controller. Although many methods and control algorithms have been explored for a quadrotor, like backstepping, sliding mode, feedback linearization, and adaptive control, the PID method was popularly applied in current control board developer [25]. The main reason of PID implementation is the simple structure, the simple hardware implementation, being easily tunable, and having a good range of performance [26]. Yet the PID controller was simple to be implemented, but the manual tuning process for its gains is a complex and time-consuming effort with required experience and caution [25]. Moreover, from the practitioner point of view, the mathematical based PID tuning is considered to be difficult and impractical for the quadrotor purpose, so they tuned their PID by *trial and error* method instead [27–30].

2.1. Quadrotor Attitude Dynamics. The quadrotor is a UAV built from four rotors installed at the corner of "plus" shaped frame (the + configuration). Each rotor consists of a fixed-pitch propeller mounted to the brushless DC motor for lifting the vehicle while controlling its attitude simultaneously. As an illustration, Figure 1 showed the quadrotor built by the Computational Intelligence and Intelligent Systems (CIIS) research group of Universitas Indonesia for their research purposes.

Quadrotor gains its recent popularity from recreational, commercial, and military purpose [31]. The commercial aspect varies from filming landscapes to movies, even delivering purchased items to buyers with the aid of GPS [32]. The advanced purposes also served by quadrotor are such as real-time information gathering for disaster area [1] and becoming part of rescuing technology in SAR activities [2]. Quadrotor is also popular as a research platform in flight control areas due to the simple design and maintenance; the ability to take-off, hover, and land vertically; being inexpensive and of lower risk compared to traditional helicopter [3, 32]. The quadrotor controlling can be illustrated as shown in Figure 2.



FIGURE 1: Quadrotor UAV.

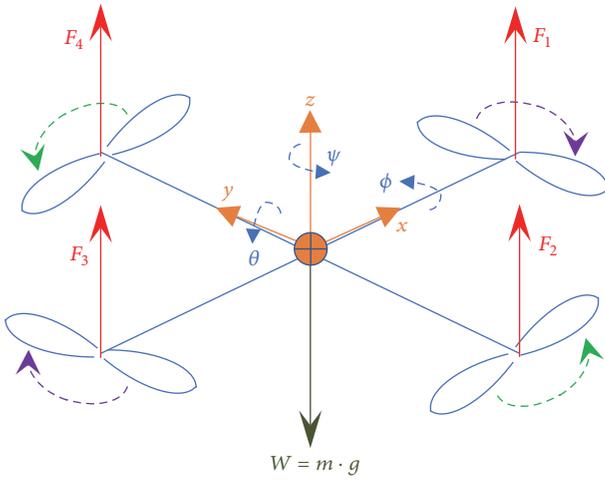


FIGURE 2: Quadrotor flight convention.

The typical quadrotor sketched in Figure 2 was having the known rotorcraft characteristics of underactuated and strong coupling in pitch-yaw-roll [33]. The latitude (z) of hovering, rolling (ϕ), pitching (θ), and yawing (ψ) are directly actuated by changing the rotor's RPM, while the forward and side translation can only be performed indirectly. This underactuated system of quadrotor dynamics will become a suitable case for control method comparison and evaluation. The addition of strong nonlinearity and coupled dynamics brought another challenge to be handled by the compared methods.

Conventionally, the quadrotor dynamic model was derived from the first principle technique. Starting with the Newton-Euler equation of motion, the rotational dynamics of UAV can be expressed by the following vector equation [34]:

$$\mathbf{J}\dot{\vec{\omega}} = \vec{u} - \vec{\omega} \times (\mathbf{J}\vec{\omega}). \quad (1)$$

Equation (1) is related to the quadrotor's angular acceleration vector, $\dot{\vec{\omega}}$, due to the control torque vector, \vec{u} , over the cross product of its current angular rate, $\vec{\omega}$, and its inertia tensor, \mathbf{J} , while the quadrotor's angular acceleration vector, $\dot{\vec{\omega}}$, is inversely proportional to the inertia tensor, \mathbf{J} . The inertia

tensor consists of three inertia terms with respect to x , y , and z rotation axis, that is, I_{xx} , I_{yy} , and I_{zz} , plus another six terms of product of inertia between their respective crossed axis, that is, I_{xy} , I_{yx} , I_{yz} , I_{zy} , I_{zx} , and I_{xz} , arranged as follows:

$$\mathbf{J} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}. \quad (2)$$

The quadrotor angular rate $\vec{\omega}$ consists of the rolling rate, p , pitching rate, q , and yawing rate, r , in the vector bracket $\vec{\omega} = [p \ q \ r]^T$. The quadrotor control torque vector, \vec{u} , consists of rolling torque, u_ϕ , pitching torque, u_θ , and yawing torque, u_ψ , as expressed in the vector bracket $\vec{u} = [u_\phi \ u_\theta \ u_\psi]^T$.

In practice, by expanding (1) together with (2), one can rearrange it to obtain another form of Newton-Euler formalism for the quadrotor attitude dynamics equation as follows [35, 36]:

$$\begin{aligned} \dot{p} &= \frac{I_{yy} - I_{zz}}{I_{xx}} qr - \frac{J_{TP}}{I_{xx}} + \frac{u_\phi}{I_{xx}} \\ \dot{q} &= \frac{I_{zz} - I_{xx}}{I_{yy}} pr - \frac{J_{TP}}{I_{yy}} + \frac{u_\theta}{I_{yy}} \\ \dot{r} &= \frac{I_{xx} - I_{yy}}{I_{zz}} p + \frac{u_\psi}{I_{zz}} \\ \ddot{z} &= -g + (\cos \theta \cos \phi) \frac{u_z}{m}. \end{aligned} \quad (3)$$

Equation (3) shows the quadrotor's angular acceleration in rolling (\dot{p}), pitching (\dot{q}), and yawing (\dot{r}) axis, as functions of moment of inertias in x -, y -, and z -axis (I_{xx} , I_{yy} , I_{zz}); rotor inertia (J_{TP}); control torque for rolling-pitching-yawing (u_ϕ , u_θ , u_ψ); and roll-pitch-yaw angular rates (p , q , r). The hovering dynamics expressed the quadrotor's vertical acceleration (\ddot{z}) as a function of mass (m), gravity constant (g), vertical control thrust (u_z), and pitching and rolling angle (θ and ϕ). In kinematics, the roll-pitch-yaw angular rates (p , q , r) were measured in body frame and rotating the quadrotor to achieve its orientation in the inertial frame. The orientation of the quadrotor can be observed as the roll-pitch-yaw flight angles (ϕ , θ , ψ) and their differential with respect to time related with p , q , r , as follows [37]:

$$\begin{Bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{Bmatrix} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \begin{Bmatrix} p \\ q \\ r \end{Bmatrix}. \quad (4)$$

Equations (3) and (4) clearly show the nonlinearity of the quadrotor angular dynamics, for example, the multiplication between states p , q , r in (1) or between states ϕ , θ , ψ in (2). The strong coupling can also be seen in the θ and ϕ interaction when moving the (\ddot{z}); in other words, to induce a vertical climb the quadrotor also experienced the pitch-roll coupling

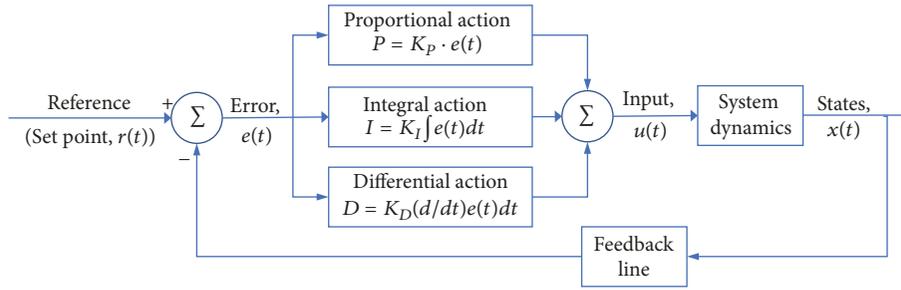


FIGURE 3: PID control scheme of universal plant system.

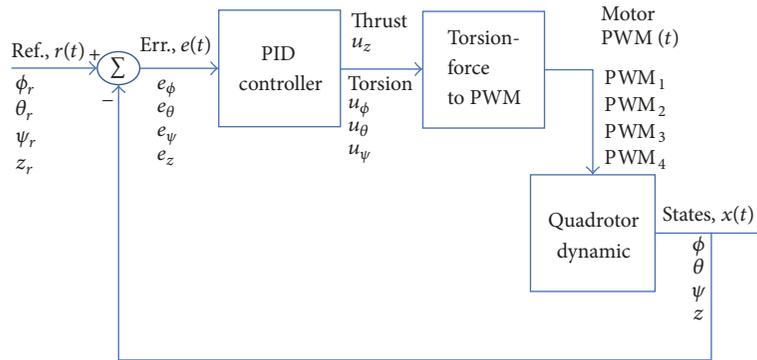


FIGURE 4: PID control scheme for quadrotor dynamic system.

and vice versa. Consequently, a proper model of altitude dynamics involved all roll, pitch, and yaw coupling each other. Thus, controlling the altitude dynamics is interrelated with the control of rolling, pitching, and yawing movement. These conditions enhanced the requirement to perform a modeling method which accommodates such strong nonlinearities. After the completion of Section 2, such requirement of nonlinear modeling is accommodated by applying the ANN-based system identification to model the quadrotor dynamics in Section 3.

2.2. The Main Ideas in PID Control Systems. The PID control method works in a feedback scheme to stabilize a system dynamics. The feedback line passes the output to the summation junction to obtain the errors and proportionally multiplies it (P-action), integrates it (I-action), and differentiates it (D-action) into the system's input to have it reduced. For optimal result, these P, I, and D actions are amplified with their respective gains, K_p , K_I , and K_D as illustrated in Figure 3.

The proportional action is speeding the system's response to counter the error according to the K_p gain. Thus, better response due to error measured should be obtained by high K_p gain. However, the increment of K_p gain above a certain limit can destabilize the system. The integral action accumulated the error and continually reduce the further error values. The optimum value of K_I eliminates the system's steady state error. The differential action processed the rate of change of the system's error. The sensitivity of this action was affected by K_D . The optimum value of K_D predicts the future error of the system, to anticipate such error.

To produce the best result, these gains must be tuned to attain the optimal combination. Such tuning was dependable on the accuracy of the dynamic modeling. For practical purposes, the developed PID technique is mostly tuned in the linearized model of the system. Thus, applying PID in quadrotor will potentially uncover its nonlinearity aspects.

2.3. Constructing PID Controller for Quadrotor Attitude Dynamics. Recently, most of the various flight controller boards like Ardupilot, Pixhawk, and OpenPilot applied the PID control method with double loop structure for roll, pitch, and yaw channel [25]. Thus, PID becomes the conventional method in UAV's autonomous feature. In the field of research, the quadrotor's PID control was processing the error values to generate the control torsion [38–40]. This is due to the application of Newton-Euler model to analyze the quadrotor flight behavior.

The controller scheme in Figure 4 compares the roll, pitch, yaw, and altitude with their references as their respective errors. Each error values were fed into the PID to produce the hovering thrust, the rolling, pitching, and yawing control torsion. Such configuration was common in quadrotor studies involving PID, according to the following structure [40–42]:

$$u_z = K_{P_z} (z_r - z) + K_{I_z} \int (z_r - z) + K_{D_z} \frac{d}{dt} (z_r - z)$$

$$u_\phi = K_{P_\phi} (\phi_r - \phi) + K_{I_\phi} \int (\phi_r - \phi) + K_{D_\phi} \frac{d}{dt} (\phi_r - \phi)$$

$$u_\theta = K_{P_\theta} (\theta_r - \theta) + K_{I_\theta} \int (\theta_r - \theta) + K_{D_\theta} \frac{d}{dt} (\theta_r - \theta)$$

TABLE I: The PID gains.

Configuration	Mode	KP	KI	KD	N	Remark
Config. #1	Roll (ϕ)	0.4	0.3	0.1	5	Assuming all rotors have similar characteristics
	Pitch (θ)	0.4	0.3	0.1	5	
	Yaw (ψ)	0.2	0.1	0.1	5	
	Altitude (z)	0.3	0.3	0.1	5	
Config. #2	Roll (ϕ)	0.4	0.3	0.1	5	Treating each rotor with its own respective coefficients
	Pitch (θ)	0.4	0.3	0.1	5	
	Yaw (ψ)	0.2	0.1	0.1	5	
	Altitude (z)	0.3	0.3	0.1	5	
Config. #3	Roll (ϕ)	2	5	2	5	Treating each rotor with its own respective coefficients
	Pitch (θ)	2	5	2	5	
	Yaw (ψ)	1	5	2	5	
	Altitude (z)	2	2.5	0.5	5	

$$\begin{aligned}
u_\psi = & K_{P_\psi} (\psi_r - \psi) + K_{I_\psi} \int (\psi_r - \psi) \\
& + K_{D_\psi} \frac{d}{dt} (\psi_r - \psi).
\end{aligned} \tag{5}$$

The symbols ϕ , θ , and ψ refer to the flight angles of roll, pitch, and yaw. Thus, u_z denotes the hovering thrust, while u_ϕ , u_θ , and u_ψ denote the rolling, pitching, and yawing torsion.

To perform the comparative simulation, the quadrotor was modeled using the ANN-based system identification which directly related the altitude and flight angles as the function of the PWM values, that is, the motor command signals. The choice of such PID structure, although common in quadrotor flight analysis was unable to be directly cascaded to the twin ANN model for comparison purpose in this article. Thus, as illustrated in previous Figure 4, a special inverse mechanism must be constructed to convert this controlling force-torsion into PWM values or the motors command signals.

2.4. Tuning the Quadrotor PID Gains . There are many types of tuning methods, varying from the simple Ziegler-Nichols to the more advanced fuzzy logic based tuning to obtain a good combination of PID gains [43]. However, an extensive study by O'Dwyer revealed that there were many industrial PID controllers “out of tune” [44]. Thus, regardless of the numerous methods available, the PID tuning was not really a simple task. Similarly, in the practical world, the quadrotor practitioners found that the tuning of quadrotor’s PID was difficult [27]. Hence, they tuned their PID by using *trial and error* method [27–30].

From their *trial and error* approach, these quadrotor practitioners developed systematics steps that differ from one to another. Beginning from the rough method, the quadrotor flight dynamics were observed, and then the PID gains are adjusted until the oscillation disappears [29] and continued with the common procedure which tuned the pitch and roll parameters with the same values and then tuned the yaw [27]. Another separated technique is tuning one axis at a time, from roll, pitch, and then yaw axis, while the quadrotor

hovers. In that axis, they adjust each gain of K_P , K_I , and K_D . Due to the coupling dynamics of quadrotor, they return to readjust the gains of previous axis [28]. The different sequence is also used by tuning of K_P first, then K_D , and finally the K_I gains [30]. Initially, the gains were set to zero and then slowly increased until the quadrotor’s oscillation critically damped; after that, the tuning process repeated until quadrotor flies in a stable manner.

A practitioner’s method in tuning quadrotor’s PID gains using computational software was reported in [45]. Although the computational software seemed to result in optimized gains, the first flight crashed due to the incorrect sampling rate, and extensive tuning had to be performed again to fly the quadrotor at the following attempt. Furthermore, the increment of the controller performance conducted by tuning the PID gains manually. These conditions have enhanced the requirement of an accurate model of quadrotor’s dynamic to correctly tune its PID gains.

The manual tuning of PID still becomes a common preferable method in quadrotor control studies [40, 41, 46]. Thus, in this work, the PID gains were manually tuned to work on the discrete form of controller [47]:

$$K_P + K_I \cdot T_s \frac{1}{z-1} + K_D \frac{N}{1 + N \cdot T_s (1/(z-1))}. \tag{6}$$

After extensive iteration, the optimal gains are obtained in Table 1.

Figure 5 showed the result of several tuning processes with the gains described in Table 1 before obtaining the preferable response of PID control on quadrotor altitude. The reference altitude profile is plotted by the green dashed line, showing two ramps and two segments of constant altitude. This choice of the pattern is explained in detail in Section 5. Thus, the objective of the tuning is to obtain such altitude response which follows the reference profile with small error to ensure the quadrotor is able to hover as commanded.

The first gain combination in the upper part of Table 1 results in the “Config 1” response drawn in black dash-dotted line in Figure 5. The gain was the same as the “Config 2” whose responses are also displayed in the blue dotted line. The significant difference between them is the way of

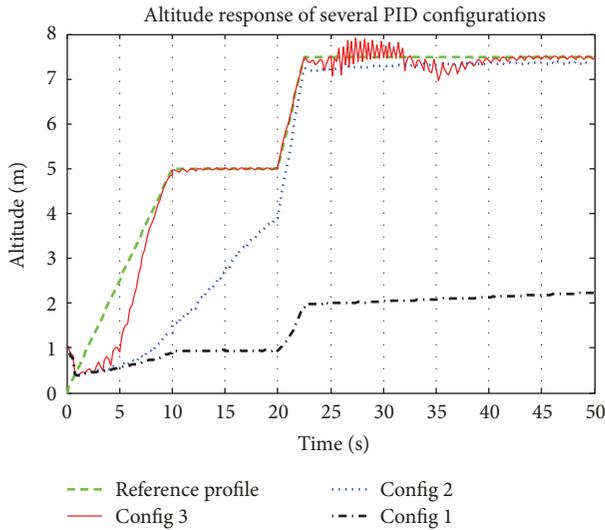


FIGURE 5: Altitude responses with respect to 3 configurations of PID tunings.

converting PID results into PWM values. In the “Config 1,” the PWMs are calculated using a single parameter of the motor, assuming that they were similar to each other. Unfortunately, each motor has their unique characteristic which cannot be neglected, and the ANN system identification captured this characteristic inside the quadrotor model. Hence, each motor then separately measured again and each of their parameters inputted the force-torsion converter to PWM. After recognizing the characteristics of each motor individually, the PID with the gain combinations in the middle part of Table 1 yield better response as plotted in “Config 2” curve. The further explanation of the PWM converter will be presented in Section 4.3.

The response of “Config 2” PID is still inadequate to bring the quadrotor into the desired altitude. Thus, the gains were increased again as in the combination in the lower part of Table 1 to speed up the response, while the limiters added and bound the controlling torsion and thrust inside the upper and lower threshold to avoid divergence. In the final trial, the “Config 3” gains obtained and enabled the quadrotor achieved the desired altitude shown with the red solid curve in Figure 5. However, a superposition of oscillation occurred as transient response. Nevertheless, since these gains are the best result of extensive trial and error process, then the PID with “Config 3” gains are chosen to be compared with the DIC-ANN in the following section. The performance of this PID controller is discussed in the comparative analysis at the Section 4.

3. Direct Inverse Control: Artificial Neural Network for Controlling Quadrotor Attitude Dynamics

This article will compare the DIC-ANN with the conventional control, which is represented by the PID method. The artificial neural network’s direct inverse control (DIC-ANN)

method itself had shown terrific achievement in controlling nonlinear dynamics aspect of UAV [19, 23, 24]. Considering the quadrotor as a highly nonlinear, strongly coupled, interacting in multivariable dynamics, and underactuated system [4], the DIC-ANN will be suitable for controlling such system. Thus, this section will present the quadrotor flight data and then discuss the DIC-ANN method together with the artificial neural-network-based modeling and control.

The direct inverse control (DIC) is a method to obtain the required input of a system to produce the desired output by constructing its dynamics inversion. In a linear system, such inversion can be performed analytically, for example, the dynamic inversion control method [48]. However, since quadrotor flight is a dynamic system with strong nonlinearity [4], the dynamic inversion will not be suitable to be implemented. In addition, the dynamic inversion cannot be performed alone for quadrotor because the system underactuated condition cannot guarantee that the vehicle will achieve the desired outputs [33]. Thus, the inversion of quadrotor’s dynamic needs to be constructed in such manner to accommodate its strong nonlinearity and underactuated condition. In this work, the DIC-ANN were implemented to control the quadrotor for a better flight performance.

3.1. The Direct Inverse Control of the ANN-Based Controller Design. The DIC-ANN applies the artificial neural network controller for a nonlinear process. Generally, the artificial neural network has been utilized for system identification and controller in wide areas of applications from the industrial nonlinear process to the vehicle control system. These included the thermal dynamic of pulsating heat pipe [49] and greenhouse temperature [50] to the autonomous vehicle control [51] and UAVs [19, 23, 24].

The DIC-ANN was constructed by directly cascading the system’s inverse, which consists of the artificial neural network controller, with the plant or the controlled system. The arrangement provides an identity that mapped the reference signal to the output of the controlled system. Consequently, the cascaded arrangement allows the artificial neural network to directly control the system’s dynamic and achieve the desired response. The cascaded arrangement illustrated in Figure 6 was developed to control the quadrotor in this article. The DIC produce required control command u , to bring the quadrotor states y , in achieving the reference states, r .

As an advantage, the DIC-ANN was able to use the most powerful feature of the artificial neural networks learning mechanism to synthesize its controller parameters. However, the initial output depended on the semirandomly initial weight matrix of the artificial neural network, which may reduce the controller robustness [52]. The block diagram illustrated in Figure 6 also showed that the direct inverse controller will depend solely on the reliability of the ANN controller in the inverse block.

In general, problems might be showed up and arisen severely because of no feedback signal involved, and furthermore, the absence of the feedback line will cause the lack of the ANN controller robustness. To optimize the open loop arrangement of the DIC, a feedback line was then equipped

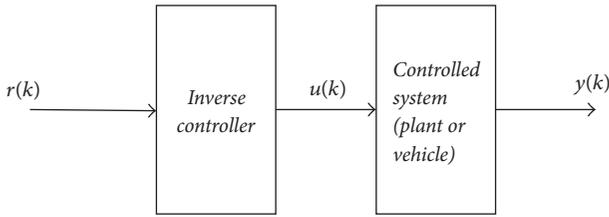


FIGURE 6: The DIC-ANN scheme.

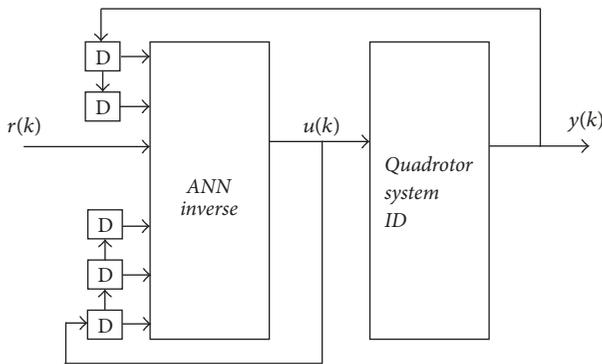


FIGURE 7: The concept of ANN inverse control cascaded with the quadrotor system identification.

into the input channel of the ANN controller so the closed loop system is achieved as illustrated in Figure 7. The required control command u , quadrotor states y , and the reference states r become the input for DIC-ANN with their delayed values symbolized by the D blocks.

The DIC applied the multilayer perceptron in its ANN controller which consists of an input layer, one hidden layer, and an output layer while the learning mechanism employed the backpropagation of error method. The learning process was conducted iteratively by adjusting the ANN weights that connect the neurons in the hidden layer with the output layer, according to the vector of the quadrotor input pattern and its desired output vector, that is, the corresponding set of the quadrotor output patterns. In parallel with the iteration, an input vector propagated forward in the network to temporarily calculate the output vector of the output layer. The difference of this output vector with its reference value will be defined as the error to be minimized by adjusting the connections weights. The backpropagation of error determines these adjustments by using the function of the neuron's mathematical model. As the learning process converges to minimize the mean square root error (MSE), the iteration stopped once the MSE fall below a chosen threshold.

3.2. The Quadrotor Telemetry Data. The artificial neural network (ANN) is a suitable method for modeling nonlinear systems, for example, quadrotor's flight dynamic. To obtain the model of the system's dynamics, the ANN runs a learning algorithm which iteratively relates the output of the system with its input until the acceptable error is achieved. Thus,

to obtain the quadrotor ANN model, the flight test must be conducted in gathering the flight data.

The UAV flight test is different from the common aircraft flight test. Due to the small dimension of the quadrotor UAV, its flight test becomes less in risk and less in the effort. The popularity of quadrotor had brought many practitioners which become a skilled pilot to fly the quadrotor for required maneuvers. The commercial controller device also equipped with data logging feature to record the gyro and accelerator output. Since conducting flight test has become feasible, the system identification for quadrotor can be performed to model its dynamics directly from the logged flight data.

The scope of this article is the altitude dynamics which involved the inner loop of the quadrotor flight control. Hence, the flight data to be processed consist of the quadrotor's altitude, its Euler flight angles, and its command signal for the motors. The Euler angles were roll, pitch, and yaw angle, and the commanded motor was PWM values for all the motor. The 4 minutes of the test result was presented in the green dash-dotted thick lines in Figure 9.

While the quadrotor was hovering, the test pilot excited the control stick to oscillate the quadrotor in rolling and pitching. In the green dash-dotted thick lines in roll and pitch graph of Figure 9, these variations logged between -20 and $+20$ degrees of flight angle. Before ending the flight test, the test pilot performs circular movement to vary the yaw angle from the direction of 100 to direction 120 (both in degrees). The altitude response showed the take-off, hover, and landing in the flight test. These flight data, especially the variations recorded, identified the quadrotor dynamics system by training the ANN model. After obtaining the sophisticated model, the ANN-DIC and PID controller can be compared head to head using this twin quadrotor dynamic model.

3.3. Modeling Quadrotor Using Artificial Neural Network System Identification. The quadrotor dynamics can be modeled by many ways, such as the physical based modeling which derived from the first principle or using the system identification methods which relied on the flight data. In a conventional way, the model from the first principle depended on the accuracy of its physical properties such as dimension, weight, and moments of inertia. However, moments of inertia can only be estimated by geometrical approximation [37, 53], CAD [54, 55], or pendulum method [56]. While the estimation relied on the measurement precision, dimension, and the construction of the pendulum [57], it also suffers from unmodeled phenomena when oscillating the UAV [58]. Thus, a system identification performed to build a reliable model for quadrotor dynamics which accommodated the nonlinearity, coupled and underactuated dynamics, that is, the ANN system identification.

Artificial neural network can be envisaged as a nonlinear transformation between multidimensional input and output [59]. In this work, the implemented ANN related the output of a system as the function of its input to be processed inter-relatedly into a set of activation functions via an optimum combination of weights. The weights then were iteratively tuned by the backpropagation of error algorithm using the

difference between the models' output from the system real output to adjust the weights.

A nonlinear dynamic system, which processes input x into output y , can be approximated with the following model:

$$y(k) = f(\Phi(k), \Theta). \quad (7)$$

$y(k)$ denotes the model's output, $\Phi(k)$ denotes the regression vector, and Θ denotes the parameter vector. Choose the Nonlinear AutoRegressive networks with eXogeneous inputs (NARX) structure using the finite amount of past inputs and past outputs as expressed by the regression function as follows:

$$\Phi(k) = (x(k-1), \dots, x(k-N_x), y(k-1), \dots, y(k-N_y)), \quad (8)$$

where N_x is the maximum lag of input and N_y is the maximum lag of output.

As the learning mechanism, the back propagation of error will approximate the nonlinear mapping for the NARX identification model as illustrated in Figure 8. The artificial neural network to identify the quadrotor dynamics will be defined by the input neuron including the input and output signal with $N_x = 2$ and $N_y = 2$, followed by a hidden layer consisting of 20 neurons and 4 output neurons. Each of the input elements ($x(k-1), \dots, x(k-N_x)$ and $y(k-1), \dots, y(k-N_y)$) is multiplied with the weight v_{xnm} or v_{ynm} and summed into each neuron in the hidden layer. The index $n = 1, 2, \dots, N$ denotes the number of input neurons and $m = 1, 2, \dots, M$ denotes the number of the hidden neurons.

These summed values were then inputted into the sigmoid activation function to obtain the output of the hidden neuron $z_m(k)$. The next step is multiplying the hidden neuron $z_m(k)$ with the weights w_m , then summing them all into the sigmoid function, and yielding the output neuron $y(k)$ according to the following expression:

$$y(k) = \sum_{m=1}^M w_m \left(\frac{1 - \exp \left\{ - \left(\sum_1^{N_y} y(k-n_y) v_{ynm} + \sum_1^{N_x} x(k-n_x) v_{xnm} \right) \right\}}{1 + \exp \left\{ - \left(\sum_1^{N_y} y(k-n_y) v_{ynm} + \sum_1^{N_x} x(k-n_x) v_{xnm} \right) \right\}} \right). \quad (9)$$

The system identification from the ANN has modeled the quadrotor dynamics in the fine result as shown in Figure 9. The measured values were plotted in the green dash-dotted line, while the modeling results were displayed in the solid red line. By using the PWM logged as input, the results showed that the ANN system identification has mimicked the quadrotor dynamics. Thus, the ANN model obtained became the Quadrotor Twin ANN ID to be controlled for both PID and DIC-ANN for the comparative simulation.

3.4. Performance of Quadrotor DIC-ANN. The DIC-ANN is composed of artificial neural network structures which represented the inverse of the system dynamics after finishing a training process. Mathematically, the DIC-ANN can be expressed in the following regression function:

$$x(k) = f(x(k-1), \dots, x(k-N_x), y(k+1), \dots, y(k-N_y)). \quad (10)$$

The schematic diagram in Figure 10 illustrates the iteration to obtain the DIC-ANN with $N_x = 2$ and $N_y = 2$ as the maximum lag of the input and output. The ANN architecture consists of a three-layer network with 28 neurons at the input layer, 20 neurons in the hidden layer, and 4 neurons in the output layer. The 28-20-4 network then trained to follow the response of the quadrotor by using the backpropagation of error algorithm.

After training the inverse block, the simulation of DIC-ANN showing satisfactory results in directing the quadrotor to achieve its desired reference. By cascading the DIC-ANN block with the system identification block and using a flight profile as the references for altitude, roll, pitch, and yaw,

the DIC showed coinciding curve between simulation results with the desired values in Figure 11. The altitude, roll, pitch, and yaw responses in the red dotted line following the variation of the reference are given in the green dashed line.

The quadrotor UAV flight responses in Figure 11 have validated the DIC-ANN performance. The ability to execute curvy and oscillatory profile as their references indicated that the DIC-ANN has accommodated the strong nonlinearity and the cross-coupling of the quadrotor dynamics. The constructed inverse block has mapped the targeted quadrotor output into a proper PWM command [19]. To assess the advantages of DIC-ANN over the conventional control method, the following sections will design the comparative method and conclude the results.

4. Deriving Comparative Methods of DIC-ANN and PID

The DIC-ANN has been reported as a promising method for the UAV flight dynamics control [19]. Beginning from the application in pattern recognition, the ANN developed into intelligent control systems and furthermore to autonomous technology in the intelligent transportation system [60]. This autonomous concept aimed to be the backbone of transportation technology to overcome the human error problem and increase the safety level of vehicle operation [60]. In current transportation applications, the autonomous technology was also implemented in UAV application for traffic monitoring [61].

This condition initiates a curiosity to analyze the increment of the performance of intelligent control method from its predecessor, the conventional control method. Moreover,

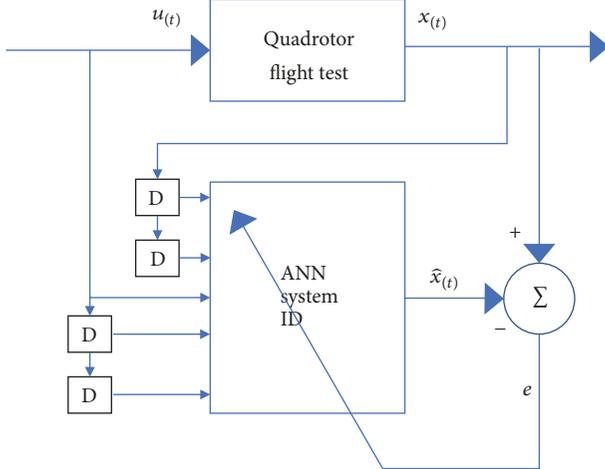


FIGURE 8: System Identification using ANN for the quadrotor dynamics.

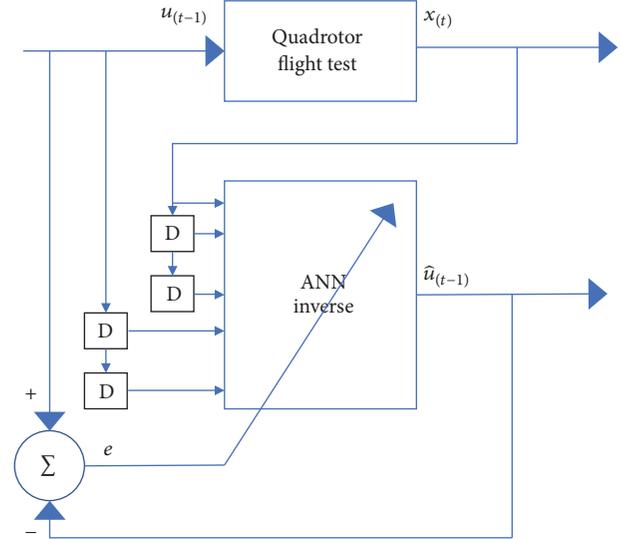


FIGURE 10: The inverse training scheme.

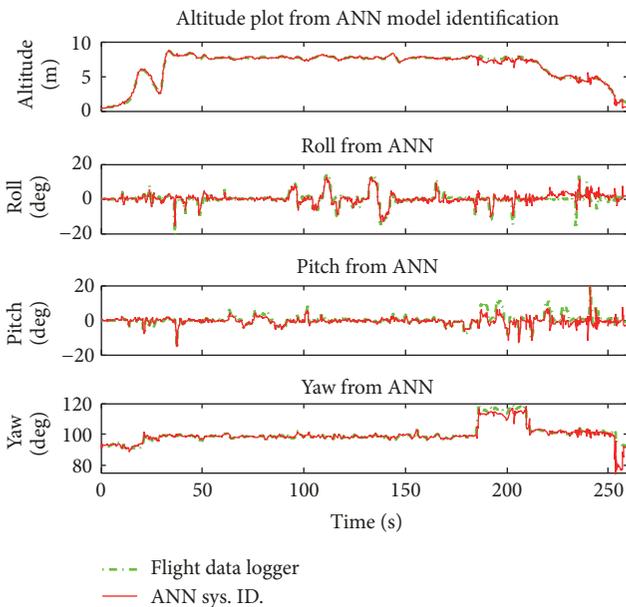


FIGURE 9: Quadrotor ANN system identification plot.

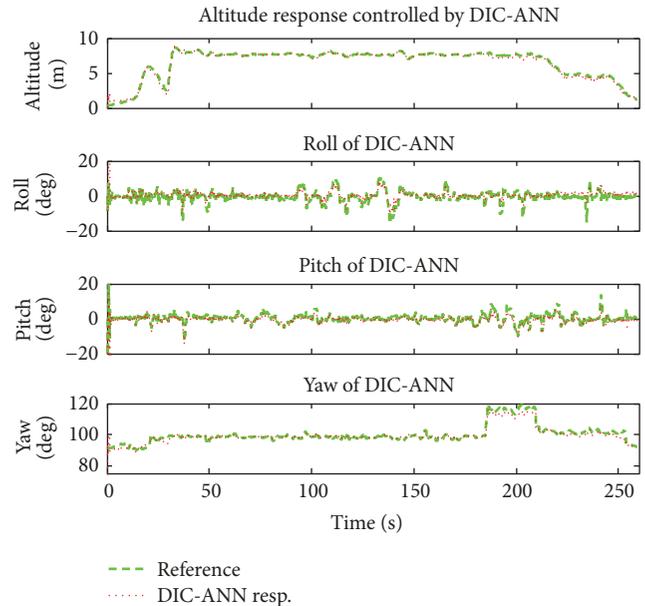


FIGURE 11: The DIC-ANN response of altitude, roll, pitch, and yaw.

the investigation of quantifying the advantage of intelligent control method over conventional method becomes a significant research. This article initiates addressing those issues by proposing a comparative methodology and, to be specific, comparatively simulating the DIC-ANN with PID control in quadrotor UAV.

4.1. The Proposed Method of Comparison. In control engineering research, there are many comparative studies between ANN-based control and conventional method that were reported, from the simple comparison of ANN control with classical control (including the comparison between ANN modeling with state space modeling) [62] to the comparison with PD control [63]. Since the PID control is widely applied, there are also several comparative studies between ANN-based control and PID at industrial purposes,

that is, in kiln [64], water bath temperature [65], compressor [66], gas production [67], and air conditioning [68].

These studies show satisfactory performance of the ANN-based method in controlling the nonlinear plants. Since quadrotor UAVs were governed in a nonlinear fashion, the implementation of ANN-based control should also be performed better than the PID. To prove such hypothesis, a proper comparative study was required in analyzing the advantages and performance increment of DIC-ANN controller from the PID.

There are many comparative studies performed between ANN-based and PID controller and hence, various methodologies are also applied. Thus, in this article, a methodology of comparison will be developed to obtain a proper conclusion.

The method to perform the comparison was designed to achieve the “apple to apple,” the “head to head,” and the “ceteris paribus” condition.

A good comparison must handle the “apple to apple” condition; that is, both control methods should be applied to the exact same quadrotor so their response will make sense to be comparable to each other. Thus, both controllers must have similar functionality in controlling the quadrotor dynamics. They must process the same type of input and produce the same type of output.

The second achievement is the “head to head” condition; that is, both control methods responses must be compared directly with each other. Consequently, both controllers will process exactly the same reference to be executed by the quadrotor. Their response will be plotted in the same graph to be directly compared and analyzed.

The last achievement is the “ceteris paribus” condition; that is, each comparison in a certain flight parameter will be held by exciting that parameter and keeping the other unexcited. The technique is applied to isolate the responses of a certain flight parameter from the rest of them.

To achieve the “apple to apple,” the “head to head,” and the “ceteris paribus” condition, the following methods are generated:

- (1) Using twin model (the ANN system identification of quadrotor dynamics) for both controllers
- (2) Converting the control force and torsion in PID output into PWM values
- (3) Simultaneously performing numerical simulation of both control methods
- (4) Using equivalent reference profile for both DIC-ANN and PID controller and then plotting their responses in a joint graph.

These steps guided the numerical simulation in producing the comparable results for analysis purposes.

4.2. Using the Quadrotor ANN System Identification as the Twin Model. The twin model was applied to obtain the “apple to apple” condition in the comparative simulation. The practice of using twin model in the joint simulation will ensure that both controllers experienced the exact same environment while interacting with one quadrotor dynamics. This type of equality will be unachievable if the comparison was held in the flight test. The controlled aerial environment will almost be impossible to be attained in the open space. Although the weather can be predicted with certain accuracy, the instantaneous wind condition is unpredicted in its direction and speed. For small dimension aircraft like UAVs, any wind occurring is not negligible for the vehicle’s dynamics. Such unexpected disturbances made the comparison be better performed in a simulation than in flight test.

Most of the studies of the PID controller applied the linearized state space model. On the other hand, most of the studies of DIC-ANN applied the ANN system identification for modeling the quadrotor dynamics. However, the use

of state space model will eliminate the nonlinear aspect which is the challenge to overcome. Thus, the ANN system identification becomes a suitable model of quadrotor dynamics for both control methods to be compared. The ANN model captured the strong nonlinearity, highly coupled, and underactuated system of quadrotor dynamics. In previous studies, besides the quadrotor, the comparison of ANN control and PID was using the ANN mutual model due to the strong nonlinearity in their plant [64, 68].

In the previous section, the quadrotor dynamics for the comparative simulation has been modeled with the ANN system identification scheme. The previous studies which modeled the UAV by the ANN system identification had proved the model’s high accuracy due to the very small value of mean of squared error [19, 23, 24]. The quadrotor ANN model in this comparative simulation receives PWM values as input and produces the roll, pitch, yaw, and altitude as the output. Thus, the ANN model becomes a more realistic representation of the quadrotor controller since the model directly processed command values to its final states which accommodate all aspects of nonlinearities and coupled dynamics to the internal mechanical interactions.

4.3. Converting Force-Torsion into PWM Signal. The quadrotor in this work was flown by four propellers and each of them was driven by BLDC motor. These motors receive command in the form of PWM signals from the main controller. These PWM signals were recorded in the flight data log and then together with the roll, pitch, yaw, and altitude were used to build the common ANN dynamic model of the quadrotor.

Consequently, a more realistic simulation can be achieved by using the PWM quantities as the controller’s output to be fed into the quadrotor dynamics model. Such arrangement was applied in directly converting the PID’s output to be PWM quantities without calculating the control forces and torsion [46]. However, such arrangement cannot be implemented here, since the output of PID was the control force and torsion. Thus, an inverse mechanism was required to convert these control force and torsion to be individual motor thrust and then invert the thrust to the motor’s PWM quantities.

In this article, the inverter mechanism was named as the “Torsion-Force-PWM converter” to denotes its input, output, and function. Similar PWM generator block was known with different terms since they have a different construction from each other, such as *rotational speed inverter* [11], *inverted movement matrix* [69], *PPM converter* [70], or *RPM-PWM mapping function* [42]. The steps of generating PWM values were summarized as follows:

- (1) Convert the control force-torsion to individual rotor thrust.
- (2) Convert the rotor thrust into its PWM values.

The first step is converting the control force-torsion to individual rotor thrust. The controlling torsion can be derived

and showed from each motor thrust by using Figure 2 in these relations:

$$\begin{aligned} u_z &= F_1 + F_2 + F_3 + F_4 \\ u_\phi &= \ell(-F_2 + F_4) \\ u_\theta &= \ell(-F_1 + F_3) \\ u_\psi &= c(F_1 - F_2 + F_3 - F_4). \end{aligned} \quad (11)$$

The notations F_1, F_2, F_3, F_4 are the thrust of motor numbers 1 to 4, respectively, ℓ are the distance between propeller's center hub and the quadrotor center point, and c is the constant of drag moment. Hence, each motor thrust can be computed as follows:

$$\begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{Bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & -\ell & 0 & \ell \\ -\ell & 0 & \ell & 0 \\ c & c & c & c \end{bmatrix}^{-1} \begin{Bmatrix} u_z \\ u_\phi \\ u_\theta \\ u_\psi \end{Bmatrix}. \quad (12)$$

The second step is converting the rotor thrust into the motor's PWM values. This conversion was obtained by the aid of polynomial regression to invert the PWM command values as a function of its thrust. The regression curve in Figure 12 was obtained after measuring the rotor's thrust in a bench test. The similar description about the thrust measurement can be viewed in [71, 72].

To obtain the c constant, the coefficient for rotor's drag moment as a function of its thrust, the measurement result was mapped in a linear curve. By using the tachometer, the rotor thrust can be fitted with a linear curve when plotted with the squared values rotor's speed, Ω^2 , in Figure 13 and continued with fitting the drag moment with the squared values rotor's speed, Ω^2 , in Figure 14. Thus, by combining their gradient, the c constant can be obtained.

Thus, the Torsion-Force-PWM converter generates the PWM in the linear approximation. After finishing the Torsion-Force-PWM converter, the PID gains can be tuned in the conventional technique. However, the plots of measurement results were showing the existence of nonlinear effect in the small amount. While the "Torsion-Force-PWM converter" excludes the existence of this nonlinearity, the DIC-ANN embedded them.

4.4. Simultaneous Numerical Simulation. Parallel with the aim of using the twin model, the simultaneous simulation also performed to ensure an "apple to apple" comparison. Thus, the simulation should run in the discrete time domain with the equivalent reference profile, each into the twin model. In simultaneous simulation it was meant that each control method runs on its own model, which is an equal model of quadrotor dynamics. The simultaneous simulation treats both control methods with an equal condition; that is, both are isolated from disturbance or environmental effect. Figure 15 showed the illustration of this simultaneous simulation.

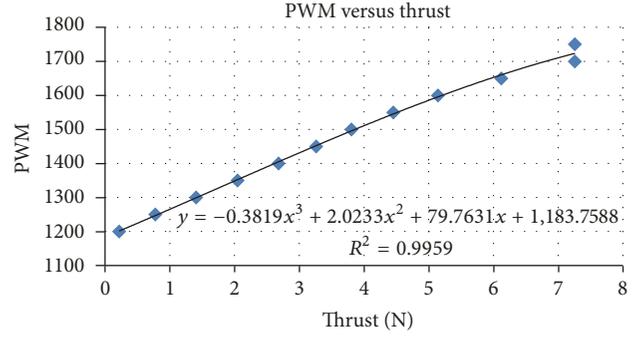


FIGURE 12: Regression of PWM from thrust.

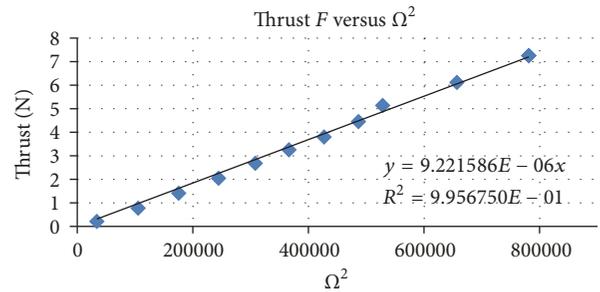


FIGURE 13: Regression of thrust with respect to squared speed.

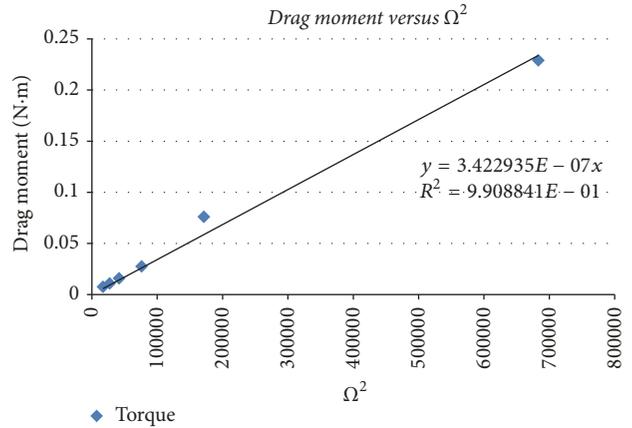


FIGURE 14: Regression of drag moment with respect to squared speed.

Using the equivalent reference profile, into the twin model, for the simultaneously numerical simulation there is a series of method designs to ensure the "apple to apple" comparison. In this work, the input was applied to the altitude channel in the ramp and constant height profile. This combination of reference profiles was designed to excite the responses of DIC-ANN and PID controller at the transient and steady condition.

Although there are many studies that compared the PID and ANN-based control systems, most of them are not directly comparing the results. Each controller's performance was plotted in the separate graph and separately analyzed [64, 65]. However, a direct comparison will produce a sharp analysis and unambiguous results. Thus, by placing the

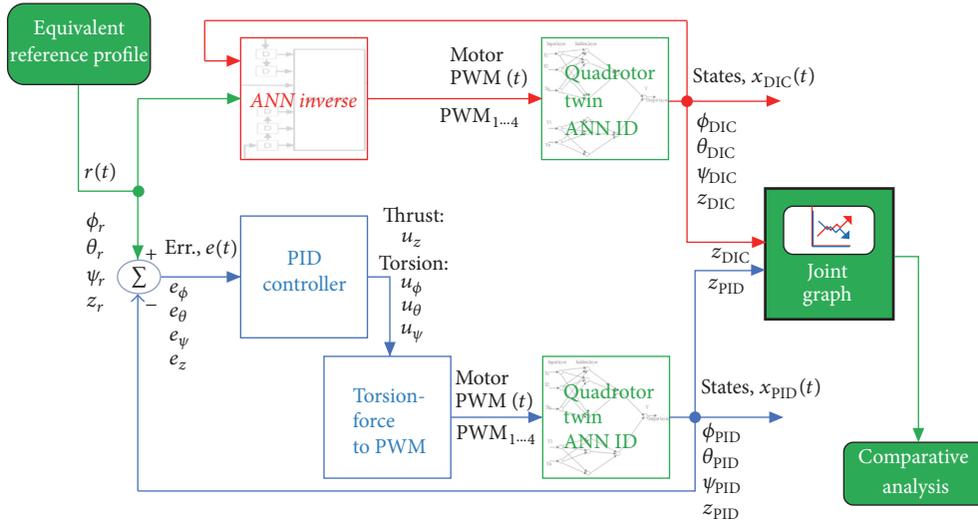


FIGURE 15: The DIC-ANN and PID comparative simulation scheme.

results of DIC-ANN and PID controller in the, respectively, mutual plot, they will be directly compared and identify their advantages or disadvantages to each other.

5. Comparison Result of DIC-ANN and PID Control Method

The comparison of DIC-ANN and PID control method is performed in the following description: first, each controller is using the twin model, second, for PID, the controlling force and torsion converted to obtain PWM values, third, the numerical simulation was executed simultaneously, and fourth, both controllers were excited with equivalent altitude profile as their reference and the responses plotted in joint graphs. Since the early descriptions have been revealed, this section compared the results in “head to head” fashion. Several published studies analyzed their comparison in the joint graph [62, 67, 68], as adopted in the following analysis.

5.1. Flight Profile for the Altitude Response Comparison. The flight simulation aimed to mimic the flight test of a quadrotor as illustrated in Figure 16. The simulated flight consists of four phases, first, the “take-off and climb” phase at $0 < t < 10$ s, second, the “hovering” phase at $10 < t < 20$ s, third the “climb in ramp” phase at $20 < t < 22.5$ s, and fourth, the “final altitude” phase at $22.5 < t < 50$ s. Initially, the quadrotor is simulated to take-off from the ground and climb to a safety altitude preferable for a test pilot to execute required maneuvers. In the numerical quantities, the quadrotor is simulated to fly from zero to 5 m of height in the “take-off and climb” phase and then “hovering” to maintain the 5 m of height. Such flight profile had enabled both controllers to achieve a steady hovering condition; thus, an equivalent excitation can be induced to stimulate the time response especially in the altitude to be observed. The excitation was enforced by lifting the quadrotor in the “climb in ramp” phase to achieve the 7.5 m height of “final altitude” phase. The slope in the “climb in ramp” phase is 1 m/s to give a 2.5 m of altitude

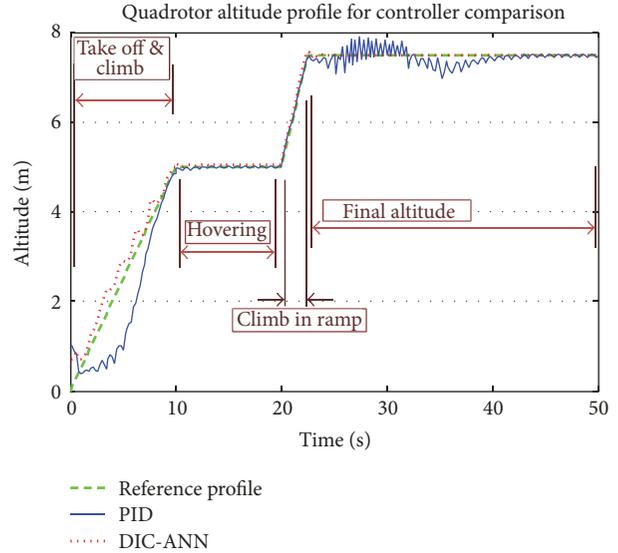


FIGURE 16: Flight profile for altitude’s response comparison.

increment from $t = 20$ s to $t = 22.5$ s. Finally, the simulation stopped at $t = 50$ s after the quadrotor altitude achieved the steady state of the “final altitude” phase.

Although, in the first 10 seconds of simulation, the DIC-ANN showed a remarkable response in following the take-off profile, no further comparison will be studied in this segment since the maneuver was meant to bring the quadrotor to the safety altitude. The comparisons are started after $t = 20$ s when the hovering quadrotor was excited by the 1m/s of ramp. The comparisons also covered the transient response at $t = 22.5$ s and the steady state response right after the ramp ended at the 7.5 m of constant height.

5.2. Comparison of the Ramp Response and Steady State. The ramp maneuver was chosen to obtain a more realistic simulation of flight compared to the standard step function

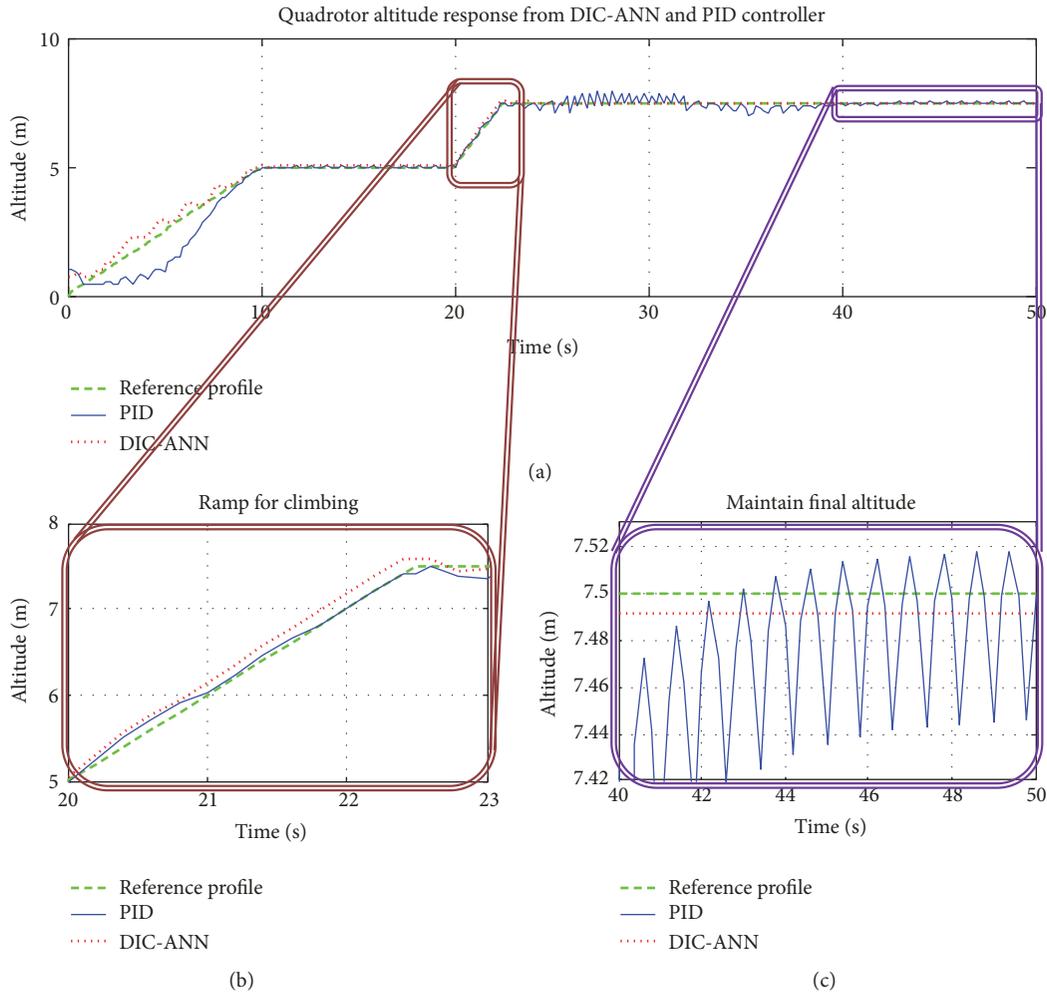


FIGURE 17: Comparison of response in ramp and steady state.

for exciting the modes of quadrotor motion. The ramp response of quadrotor altitude is shown in Figure 17(b) entitled “ramp for climbing.” The steady state response was shown in Figure 17(c) entitled “maintain final altitude.” The reference altitude profile is displayed as the green dashed line, and the DIC-ANN response was represented by the red dotted curve while the PID response was presented by the solid blue curve.

Along with the reference profile, the DIC-ANN altitude response was quickly damped and rising to follow the reference slope without significant oscillation observed. While the oscillation in PID controller response was observed for 2.5 s until the ramp ended. A closer examination in the Figure 17(b) showed that the PID controller produced a lower error when following the ramp maneuver compared to the DIC-ANN altitude response.

After the ramp completed at $t = 22.5$ s and becomes constant altitude in the reference profile, the altitude response shows the transient oscillatory mode until it damped and achieved the steady state condition. The steady state achievement can be seen in Figure 17(c), while the transient response is discussed in the following subsection. Although the

simulation runs 37.5 s after the transient occurrence at the edge of the ramp, the PID controller is still unable to damp the oscillatory mode. This residual oscillatory mode forms the PID resulting in a vertical vibration with 0.04 m or 4 cm of amplitude and 1 Hz of frequency, that is, significant enough with the quadrotor dimension and its potential mission performance. On the contrary, the DIC has a shorter transient response and achieves its steady state for 3 s after the ramp is completed, shown in Figure 18(b) at the following subsection. Thus, the DIC-ANN has settling time equal to 3 s while the PID controller fails to settle after 37.5 s due to the residual oscillatory mode with 0.04 m or 4 cm of amplitude and 1 Hertz of frequency.

5.3. Comparison of the Transient Responses. The quadrotor was brought up from 5 m of altitude to 7.5 m by a 1 m/s ramp maneuver. Hence, the ramp is completed after 2.5 s and the altitude is being constantly maintained. However, a sudden discontinuity at the end of ramp maneuver excites the oscillatory mode of the quadrotor flight motion. These oscillatory motions, known as the transient response, occurred at both of DIC-ANN and PID controllers. The transient

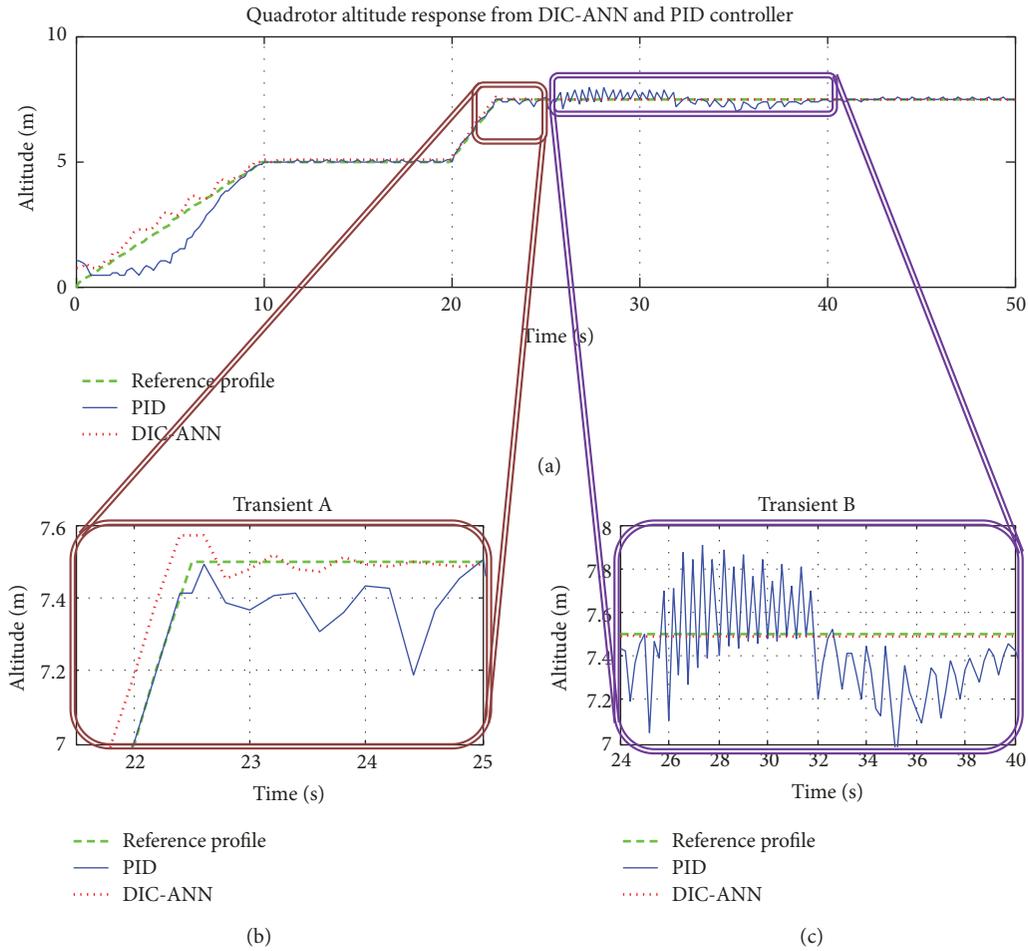


FIGURE 18: Comparison of the transient response.

responses of the quadrotor altitude are displayed in Figures 18(b) and 18(c). Although the DIC-ANN damped the transient response immediately, the PID controller still exhibits its transient response involving multiple frequencies. These plural frequencies were exposed by splitting the display into two sequential graphs, that is, Figure 18(b) entitled “Transient A” for $21.5\text{ s} < t < 25\text{ s}$ and Figure 18(b) entitled “Transient B” for $24\text{ s} < t < 40\text{ s}$. The reference altitude profile is depicted as the green dashed line, and the DIC-ANN response is represented by the red dotted curve while the PID response is represented by the blue solid curve.

As interpreted from Figure 18(b), the DIC-ANN successfully damped the transient response and brought the quadrotor altitude to its desired height. On the other hand, the PID experienced 1 Hz transient oscillation which was undamped after 3 s. As the observation continued, the PID controller in Figure 18(c) showed the superposition of the short period (large frequencies) oscillation modulated with the long period (rare frequencies), for example, the swing with 16 s period. To be concluded, the DIC-ANN have better performance in handling the transient response of quadrotor altitude compared to the PID controller, since DIC-ANN damped the oscillation after 3 s while PID was unable to do so for 17.5 s after excitation occurred.

The absolute error of both controllers presented in Figures 19(a), 19(b), and 19(c), while Figure 19(d) showed the altitude profile to relate itself with the corresponding error characteristic. The DIC-ANN absolute error is represented by the red dash-dotted curve while the PID absolute error is represented by the diamond-marked blue curve. The absolute error of ramp execution displayed in Figure 19(b) provides the maximum error of DIC-ANN which does not exceed 0.2 m. After $t = 22\text{ s}$, the DIC-ANN absolute error tends to be reduced as shown in the slight decrement which ended at $t = 22.5\text{ s}$ when the ramp is completed. Since most of the quadrotor mission required altitude precision in the hovering maneuver, thus, the DIC-ANN ramp errors are still acceptable although the PID have smaller error, i.e., a half of the DIC-ANN. On the contrary, Figure 19(c) highlighted the PID absolute error to 0.5 m from the desired altitude in the final altitude aimed. After the quadrotor completed the ramp, the DIC-ANN ramp error decreased immediately in less than 3 s as shown in Figure 19(c) while the PID ramp error increased and oscillated between 0 and 0.5 m until 17.5 s after the ramp completed. The absolute error characteristic enhanced the DIC-ANN capability to overcome a sudden change in the reference profile and following it with lower oscillation than the PID.

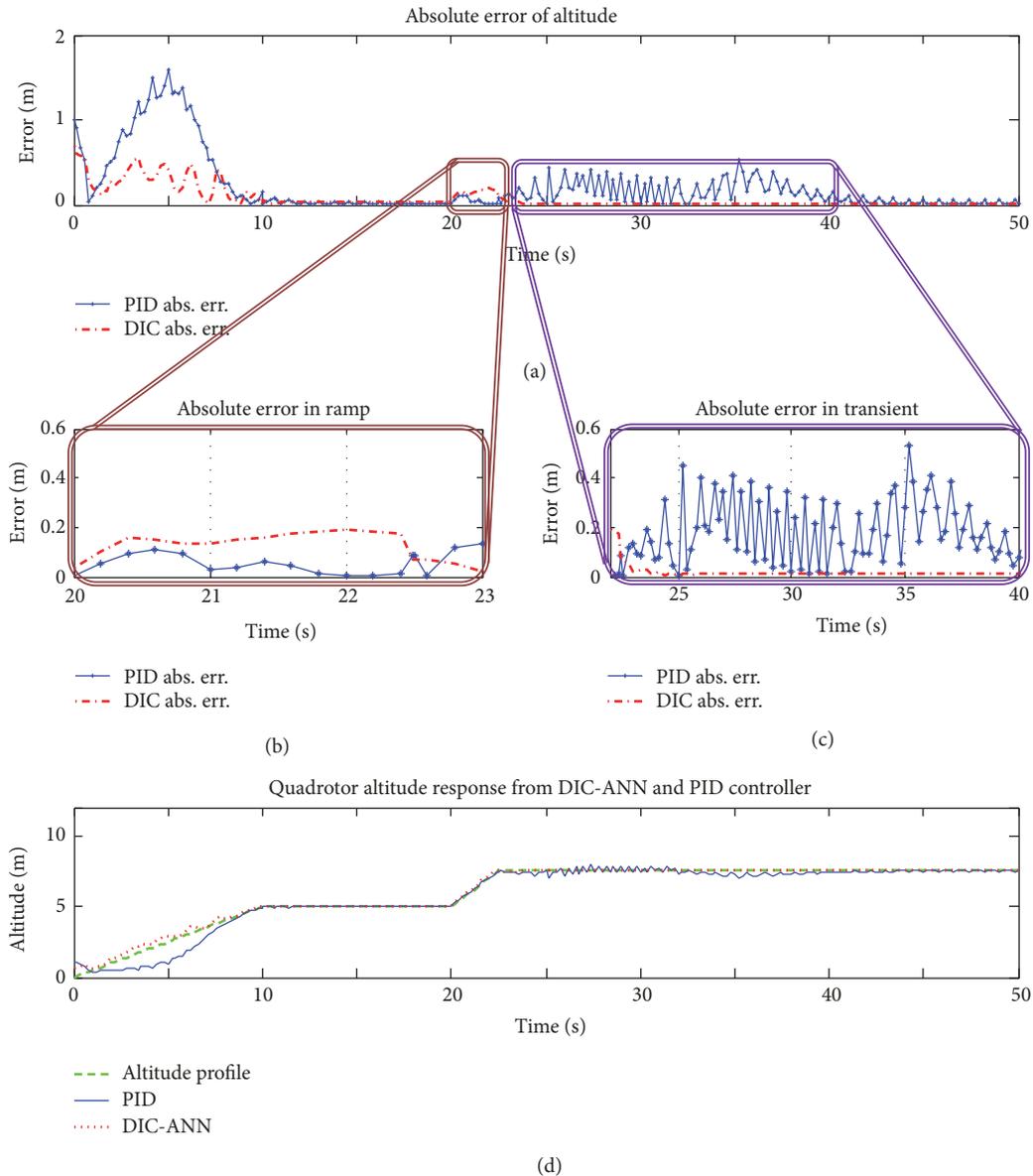


FIGURE 19: Absolute error of DIC-ANN and PID in altitude response.

6. Concluding Remarks

The method of comparison built in this article originated from the satisfactory results of DIC-ANN controller in UAVs. Thus, the gap in the comparative research between DIC-ANN and PID for quadrotor has been elaborated to pull out a proper comparative method between these control systems. The comparison composed the twin ANN model usage, the conversion into PWM values, the simultaneous numerical simulation, and the equivalent reference profile to be plotted on a joint graph. These steps ensured the comparison to be *apple to apple, head to head, and ceteris paribus*.

The comparison result showed that the DIC-ANN performance was better than the PID controller in handling the quadrotor altitude dynamics. Although the DIC experienced

slight overshoot in following the ramp input given, the error is not exceeding 0.2 m while it tends to be decreased.

The sudden changes from climbing to hover exhibit the lower steady state error in the DIC-ANN controller and the transient oscillation damped faster by the DIC-ANN. These results showed that DIC-ANN was better in controlling the nonlinear vehicle than the PID controller did. Equipped with the learning algorithm, the DIC-ANN performed better compared to the PID that involved efforts of tuning. Such results will support the DIC-ANN application for the autonomous technology. For further development, the comparison will be suitable to be expanded to analyze the translation responses. By analyzing the UAV responses in the *x* and *y* direction, the trajectory can be compared and validated.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

- [1] T. Prucksakorn, K. Wachirattanakornkul, and I. Nilkhamhang, "Unmanned aerial vehicle for observing landslide with iterative feedback tuning," in *Proceedings of the 2013 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, ECTI-CON 2013*, May 2013.
- [2] A. Sargent, Ed., *Drone tested by Albany sea rescuers at Salmon Holes set to help in emergencies*, ABC News, 2017.
- [3] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin, "Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment," in *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, Hilton Head, South Carolina, 2007.
- [4] X. Zhang, X. Li, K. Wang, and Y. Lu, "A survey of modelling and identification of quadrotor robot," *Abstract and Applied Analysis*, vol. 2014, Article ID 320526, 2014.
- [5] J. Mange, S. Pace, and A. Dunn, "Optimization of PID Controller Parameters for Automated Ground Vehicle Control on Dynamic Terrain," in *International Conference on Scientific Computing (CSC)*, p. 3, Las Vegas, Nev, USA, 2015.
- [6] C. S. Chin, M. W. S. Lau, and E. Low, "Supervisory cascaded controller design: experiment test on a remotely operated vehicle," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 225, no. 3, pp. 584–603, 2011.
- [7] C. S. Chin, M. W. S. Lau, E. Low, and G. G. L. Seet, "Robust and decoupled cascaded control system of underwater robotic vehicle for stabilization and pipeline tracking," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 222, no. 4, pp. 261–278, 2008.
- [8] R. Cui, C. Yang, Y. Li, and S. Sharma, "Adaptive Neural Network Control of AUVs With Control Input Nonlinearities Using Reinforcement Learning," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 6, pp. 1019–1029, 2017.
- [9] R. Cui, L. Chen, C. Yang, and M. Chen, "Extended state observer-based integral sliding mode control for an underwater robot with unknown disturbances and uncertain nonlinearities," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 8, pp. 6785–6795, 2017.
- [10] C. Tian, J. Wang, Z. Yin, and G. Yu, "Integral backstepping based nonlinear control for quadrotor," in *Proceedings of the 35th Chinese Control Conference, CCC 2016*, pp. 10581–10585, July 2016.
- [11] Q. Wei, H. Wang, Q. Wu, and M. Chen, "Backstepping-based attitude control for a quadrotor UAV using nonlinear disturbance observer," in *Proceedings of the 34th Chinese Control Conference, CCC 2015*, pp. 771–776, July 2015.
- [12] D. Matouk, O. Gherouat, F. Abdessemed, and A. Hassam, "Quadrotor position and attitude control via backstepping approach," in *Proceedings of the 8th International Conference on Modelling, Identification and Control, ICMIC 2016*, pp. 73–79, November 2016.
- [13] Y. Yu, Y. Guo, X. Pan, and C. Sun, "Robust backstepping tracking control of uncertain MIMO nonlinear systems with application to quadrotor UAVs," in *Proceedings of the 2015 IEEE International Conference on Information and Automation, ICIA 2015 - In conjunction with 2015 IEEE International Conference on Automation and Logistics*, pp. 2868–2873, August 2015.
- [14] M. Reinoso, L. I. Minchala, J. P. Ortiz, D. Astudillo, and D. Verdugo, "Trajectory tracking of a quadrotor using sliding mode control," *IEEE Latin America Transactions*, vol. 14, no. 5, pp. 2157–2166, 2016.
- [15] M. K. Shaik and J. F. Whidborne, "Robust sliding mode control of a quadrotor," in *Proceedings of the 11th UKACC United Kingdom Automatic Control Council International Conference on Control, UKACC Control 2016*, September 2016.
- [16] J. Anwar and F. M. Malik, "Performance of non-linear observers for closed loop terminal sliding mode control of quadrotor UAV," in *Proceedings of the 2017 14th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, pp. 244–252, Islamabad, Pakistan, January 2017.
- [17] P. Dey, S. R. Kurode, and R. Ramachandran, "Robust attitude control of quadrotor using sliding mode," in *Proceedings of the 1st International Conference on Automatic Control and Dynamic Optimization Techniques, ICACDOT 2016*, pp. 268–272, September 2016.
- [18] L. Li, L. Sun, and J. Jin, "Survey of advances in control algorithms of quadrotor unmanned aerial vehicle," in *Proceedings of the 16th IEEE International Conference on Communication Technology, ICCT 2015*, pp. 107–111, October 2015.
- [19] B. Kusumoputro, H. Suprijono, M. A. Heryanto, and B. Y. Suprpto, "Development of an attitude control system of a heavy-lift hexacopter using Elman recurrent neural networks," in *Proceedings of the 22nd International Conference on Automation and Computing, ICAC 2016*, pp. 27–31, September 2016.
- [20] Y. Li and S. Song, "A survey of control algorithms for quadrotor unmanned helicopter," in *Proceedings of the IEEE 5th International Conference on Advanced Computational Intelligence (ICACI '12)*, pp. 365–369, October 2012.
- [21] E. R. Rene, M. E. López, J. H. Kim, and H. S. Park, "Back propagation neural network model for predicting the performance of immobilized cell biofilters handling gas-phase hydrogen sulphide and ammonia," *BioMed Research International*, vol. 2013, Article ID 463401, 2013.
- [22] D. Sutarya and B. Kusumoputro, "Identification of industrial furnace temperature for sintering process in nuclear fuel fabrication using NARX neural networks," *Science and Technology of Nuclear Installations*, vol. 2014, Article ID 854569, 8 pages, 2014.
- [23] H. Suprijono and B. Kusumoputro, "Direct inverse control based on neural network for unmanned small helicopter attitude and altitude control," *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 9, no. 2-2, pp. 99–102, 2017.
- [24] M. A. Heryanto, H. Suprijono, B. Y. Suprpto, and B. Kusumoputro, "Attitude and Altitude control of a quadcopter using neural network based direct inverse control scheme," *Advanced Science Letters*, vol. 23, no. 5, pp. 4060–4064, 2017.
- [25] A. Julkananusart and I. Nilkhamhang, "Quadrotor tuning for attitude control based on double-loop PID controller using fictitious reference iterative tuning (FRIT)," in *Proceedings of the 41st Annual Conference of the IEEE Industrial Electronics Society, IECON 2015*, pp. 4865–4870, November 2015.
- [26] C. K. Tan and J. Wang, "A novel PID controller gain tuning method for a quadrotor landing on a ship deck using the invariant ellipsoid technique," in *Proceedings of the 2014 14th*

- International Conference on Control, Automation and Systems, ICCAS 2014*, pp. 1339–1344, October 2014.
- [27] M. marc81, “Quadcopter PID output, Message to Robotics Stack Exchange Online Discussion Forum,” 2014, <http://robotics.stackexchange.com/questions/2964/quadcopter-pid-output/2965#2965>.
- [28] O. Liang, “Quadrotor PID Explained and Tuning, Share Knowledge and Ideas, March 19,” <http://oscarliang.com/quadcopter-pid-explained-tuning/>.
- [29] E. H. T. Teisberg, “Scratch-Built Quadcopter Project, Thomas Teisberg, March 19,” <http://www.thomasteisberg.com/quadcopter/>.
- [30] hauptmech. and J. Baker, “What are good strategies for tuning PID loops?, Message to Robotics Stack Exchange Online Discussion Forum, 2012,” <http://robotics.stackexchange.com/questions/167/what-are-good-strategies-for-tuning-pid-loops>.
- [31] R. Allen and M. Pavone, “A Real-Time Framework for Kinodynamic Planning with Application to Quadrotor Obstacle Avoidance,” in *AIAA Guidance, Navigation, and Control Conference*, Gaylord Palms, Kissimmee, Florida, USA, 2016.
- [32] K. N. Dang, G. Lee, and T. Kang, “Linear quadrotor modelling and attitude controller design based on experimental data,” in *Proceedings of the 15th International Conference on Control, Automation and Systems, ICCAS 2015*, pp. 472–476, October 2015.
- [33] A. Das, K. Subbarao, and F. Lewis, “Dynamic inversion with zero-dynamics stabilisation for quadrotor control,” *IET Control Theory & Applications*, vol. 3, no. 3, pp. 303–314, 2009.
- [34] A. Honglei, L. Jie, W. Jian, W. Jianwen, and M. Hongxu, “Backstepping-based inverse optimal attitude control of quadrotor,” *International Journal of Advanced Robotic Systems*, vol. 10, article no. 223, 2013.
- [35] S. Bouabdallah and R. Siegwart, “Full control of a quadrotor,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '07)*, pp. 153–158, San Diego, Calif, USA, November 2007.
- [36] T. Bresciani, *Modelling, identification and control of a quadrotor helicopter [M.S. theses]*, 2008.
- [37] R. Beard, “Quadrotor dynamics and control, All Faculty Publications, 2008,” <http://scholarsarchive.byu.edu/facpub/1325>.
- [38] H. Zheng, Q. Zeng, W. Chen, H. Zhu, and C. Chen, “Improved PID control algorithm for quadrotor based on MCS,” in *Proceedings of the 7th IEEE Chinese Guidance, Navigation and Control Conference, CGNCC 2016*, pp. 1780–1783, August 2016.
- [39] J. O. Pedro, M. Dangor, and P. J. Kala, “Differential evolution-based PID control of a quadrotor system for hovering application,” in *Proceedings of the 2016 IEEE Congress on Evolutionary Computation, CEC 2016*, pp. 2791–2798, July 2016.
- [40] K. Issam and G. Qingbo, “Research on control strategies for the stabilization of quadrotor UAV,” in *Proceedings of the 5th International Conference on Intelligent Control and Information Processing, ICICIP 2014*, pp. 286–292, August 2014.
- [41] M. Elsamanty, A. Khalifa, M. Fanni, A. Ramadan, and A. Abo-Ismael, “Methodology for identifying quadrotor parameters, attitude estimation and control,” in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM '13)*, pp. 1343–1348, Wollongong, Australia, July 2013.
- [42] C. Wang, M. Nahon, and M. Trentini, “Controller development and validation for a small quadrotor with compensation for model variation,” in *Proceedings of the 2014 International Conference on Unmanned Aircraft Systems, ICUAS 2014*, pp. 902–909, May 2014.
- [43] M. Reyad, M. Arafa, and E. A. Sallam, “An optimal PID controller for a quadrotor system based on de algorithm,” in *Proceedings of the 11th International Conference on Computer Engineering and Systems, ICCES 2016*, pp. 444–451, December 2016.
- [44] A. O’Dwyer, “An Overview of Tuning Rules for the PI and PID Continuous-Time Control of Time-Delayed Single-Input, Single-Output (SISO) Processes,” in *PID Control in the Third Millennium, Advances in Industrial Control*, pp. 3–44, Springer, London, UK, 2012.
- [45] A. Mukarram, U. A. Fiaz, and U. I. Khan, *Altitude Control of a Quadcopter*, Department of Electrical Engineering, Pakistan Institute of Engineering and Applied Sciences, Islamabad, Pakistan, 2015.
- [46] G. Bo, L. Xin, Z. Hui, and W. Ling, “Quadrotor helicopter Attitude Control using cascade PID,” in *Proceedings of the 28th Chinese Control and Decision Conference, CCDC 2016*, pp. 5158–5163, May 2016.
- [47] Mathworks., “PID Controller, Discrete PID Controller,” <https://www.mathworks.com/help/simulink/slref/pidcontroller.html>.
- [48] B. L. Stevens, F. L. Lewis, and E. N. Johnson, *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems*, John Wiley and Sons, Inc, 3rd edition, 2015.
- [49] A. Jokar, A. A. Godarzi, M. Saber, and M. B. Shafii, “Simulation and optimization of a pulsating heat pipe using artificial neural network and genetic algorithm,” *Heat and Mass Transfer*, vol. 52, no. 11, pp. 2437–2445, 2016.
- [50] A. Manonmani, T. Thyagarajan, M. Elango, and S. Sutha, “Modelling and control of greenhouse system using neural networks,” *Transactions of the Institute of Measurement and Control*, 2016.
- [51] P. Sinha, L. Hadjiiski, and K. Mutib, “Neural Networks in Autonomous Vehicle Control,” *IFAC Proceedings Volumes*, vol. 26, no. 1, pp. 335–340, 1993.
- [52] B. Kusumoputro, K. Priandana, and W. Wahab, “System identification and control of pressure process rig® system using back-propagation neural networks,” *ARNP Journal of Engineering and Applied Sciences*, vol. 10, no. 16, pp. 7190–7195, 2015.
- [53] R. Syam, “Dynamics and fuzzy logic method for controlling quadcopter,” *Research Journal of Applied Sciences*, vol. 11, no. 6, pp. 251–260, 2016.
- [54] W. Hussein, M. El-khatib, A. Elruby, and H. Haleem, “Quad Rotor Design, Simulation And Implementation,” in *International Conference on Computer Science from Algorithms to Applications, (CSAA09)*, JW Marriott, Mirage City, Cairo, Egypt, 2009.
- [55] H. Oh, D.-Y. Won, S.-S. Huh, D. H. Shim, M.-J. Tahk, and A. Tsourdos, “Indoor UAV Control Using Multi-Camera Visual Feedback,” *Journal of Intelligent Robotic Systems*, vol. 61, no. 1, pp. 57–84, 2011.
- [56] J. Yang, Z. Cai, Q. Lin, and Y. Wang, “Self-tuning PID control design for quadrotor UAV based on adaptive pole placement control,” in *Proceedings of the 2013 Chinese Automation Congress, CAC 2013*, pp. 233–237, November 2013.
- [57] A. Teimourian and D. Firouzbakht, “A Practical Method for Determination of the Moments of Inertia of Unmanned Aerial Vehicles,” in *XXII Conference of Italian Association of Aeronautics and Astronautics*, Napoli, Italy, 2013.

- [58] M. R. Jardin and E. R. Mueller, "Optimized measurements of UAV mass moment of inertia with a bifilar pendulum," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, pp. 4790–4812, Monterey, Calif, USA, August 2007.
- [59] A. E. Ruano, P. M. Ferreira, and C. M. Fonseca, "An overview of nonlinear identification and control with neural networks," in *Intelligent Control Systems using Computational Intelligence Techniques, IET Control Engineering Series*, A. E. Ruano, Ed., p. 37, The Institution of Engineering and Technology, London, UK, 2008.
- [60] C.-H. Chen, "Intelligent transportation control system design using wavelet neural network and PID-type learning algorithms," *Expert Systems with Applications*, vol. 38, no. 6, pp. 6926–6939, 2011.
- [61] Y. Du, C. Zhao, F. Li, and X. Yang, "An Open Data Platform for Traffic Parameters Measurement via Multirotor Unmanned Aerial Vehicles Video," *Journal of Advanced Transportation*, vol. 2017, Article ID 8324301, 12 pages, 2017.
- [62] P. Ryckaert, B. Salaets, A. S. White, and M. Stoker, "A comparison of classical, state-space and neural network control to avoid slip," *Mechatronics*, vol. 15, no. 10, pp. 1273–1288, 2005.
- [63] M. Azizur Rahman and M. Ashraful Hoque, "On-line self-tuning ANN-based speed control of a PM DC motor," *IEEE/ASME Transactions on Mechatronics*, vol. 2, no. 3, pp. 169–178, 1997.
- [64] F. M. Dias and A. M. Mota, "A comparison between a PID and internal model control using neural networks," in *5th World Multi-Conference on Systemics, Cybernetics and Informatics*, vol. 268, p. 273, Orlando, EUA, 2001.
- [65] Y. V. Parkale, "Comparison of ANN Controller and PID Controller for Industrial Water Bath Temperature Control System using MATLAB Environment," *International Journal of Computer Applications*, vol. 53, no. 2, pp. 1–6, 2012.
- [66] M. I. Mohammed, Z. B. Noussier, R. M. Abdel Maksoud, and F. Z. Amer, "Surge avoidance using speed and valve methodologies controlled by PID, fuzzy and neural networks approaches," in *The 2011 International Conference on Computer Engineering and Systems*, pp. 73–78, IEEE, Cairo, Egypt, December 2011.
- [67] M. Y. Azwar, M. A. Hussain, A. K. Abdul Wahab, and M. F. Zanil, "A Comparative study between Neural Networks (NN)-based and Adaptive-PID Controllers for the Optimal Bio-Hydrogen Gas Production in Microbial Electrolysis Cell Reactor," *Computer Aided Chemical Engineering*, vol. 37, pp. 1529–1534, 2015.
- [68] P. Kumar and A. P. K. Singh, "Comparative Analysis of Air Conditioning System Using PID and Neural Network Controller," *International Journal of Scientific and Research Publications*, vol. 795, 2013.
- [69] D. Bokan, N. Cetic, J. Kovacevic, and V. Ilic, "Control of unmanned aerial vehicle based on XMOS platform," in *Proceedings of the 2013 21st Telecommunications Forum Telfor, TELFOR 2013*, pp. 893–896, November 2013.
- [70] K. Li, S. K. Phang, B. M. Chen, and T. H. Lee, "Platform design and mathematical modeling of an ultralight quadrotor micro aerial vehicle," in *Proceedings of the 2013 International Conference on Unmanned Aircraft Systems, ICUAS 2013*, May 2013.
- [71] P. S. King, *System Design Methodology and Implementation of Micro Aerial Vehicles*, Graduate School for Integrative Sciences and Engineering, National University of Singapore, Singapore, 2014.
- [72] M. Dupuis, J. Gibbons, M. Hobson-Dupont et al., *Design Optimization of a Quad-Rotor Capable of Autonomous Flight*, Aerospace Engineering, Worcester Polytechnic Institute, Worcester, Mass, USA, 2008.

