American Academic Scientific Research Journal for Engineering, Technology, and Sciences
ISSN (Print) 2313-4410, ISSN (Online) 2313-4402
http://asrjetsjournal.org/

Optimized PID, FOPID and PIDD^2 for Controlling UAV Based on SSA

Nagham M. Abdulridha[a*], Prof. Dr. Ali Hussien Mary[b], Asst. Prof. Dr. Hisham H. Jasim[c]

[a,b,c]*Mechatronics Department, Al-Khwarizmi College of Engineering, University of Baghdad, Baghdad 10001, Iraq*

[a]*Email: nagham.mo7@gmail.com*, [b]*Email: Alimary76@kecbu.uobaghdad.edu.iq*

[c]*Email: Hisham@kecbu.uobaghdad.edu.iq*

Abstract

Unmanned Aerial Vehicles (UAVs) are widely used in recent years for different applications. Thus, UAV control attracted many researchers to suggest suitable controllers. The simplicity of PID controller makes it the first choice. In this paper, an offline tuning procedure based on Salp Swarm Algorithm (SSA) for the attitude control of UAV is proposed. The parameters of PID, Fractional Order PID (FOPID), and PID Plus Second-Order Derivative (PIDD^2) have been tuned and their performance is compared in terms of rise time, maximum overshoot, settling time, and integral time absolute error.

Keywords: UAV; SSA; PID; FOPID; PIDD^2.

1. Introduction

Autonomous flying vehicles (UAVs) have recently aroused the interest of the commercial, industrial, and academic sectors because to the wide range of activities they can accomplish, their capacity to maneuver, and cover a large distance quickly also they can go to dangerous places where people cannot go [1]. Quadrotor is a type of the UAV with four rotors that has attracted the interest of researchers due to advantages such as vertical take-off and landing (VTOL), lower mechanical complexity, payload enhancement, gyroscopic effect reduction, practical flight modes, variety of sizes, perfect maneuvering, and less damage in the event of a collision [2]. The rotors of the quadrotor arranged in a "+" or "X" shape and is controlled by variations in motor speed; consequently, it does not have complicated mechanical control linkages. Flight control becomes a difficult problem due to the coupled dynamics, which are highly nonlinear, an under-actuated design configuration, and different uncertainties encountered during flight.

----

* Corresponding author.

Quadrotors are always subject to several disturbances and uncertainties during flight, including as parametric perturbations, noise, and wind gusts, then it is difficult to create a control to increase quadrotor flying performance: therefore, there are many researchers improved their controller (classic or intelligent controllers).

For example, Nguyen and colleagues[3] presented a control algorithm for quadrotor’s altitude which consists of a combination of nonlinear and linear controllers. Pan and colleagues[4] proposed a double-loop cascaded linear quadratic regulator (LQR) controller to track the referred position and attitude respectively, then they designed Simultaneous Perturbation Stochastic Approximation (SPSA) to optimized the parameters in the position and altitude controllers, also Okyere and colleagues[5] and Reyes-Valeria and colleagues[6] used LQR controller, while Bolandi and colleagues[7] used Proportional-Integral-Derivative (PID) controller to control the attitude of the quadrotor and optimized the parameters by Genetic Algorithm (GA). Karahan and colleagues[8] and Kamel and colleagues[9] also used PID controller, while in Argentim and colleagues[10] presented PID tuned by LQR loop, and in Sheng and colleagues[11] presented fuzzy control PID system for civil four-rotors UAV while Yin and colleagues[12] designed a double-loop controller for a quadrotor by using PID and Sliding mode controller (SMC). Also, Kara and colleagues[13, 14] combined a linear controller with non-linear controller used SMC with PID Or PD to control a trajectory tracking then the parameters designed based on Lyapunov theory.

According to the above researches, the majority of existing approaches are either difficult to create and implement or require great computational resources. Meanwhile, PID control legislation appears to be important in determining a simple and efficient control strategy for a wide range of systems. This paper is organized from the beginning of the introduction then the second section present the mathematical model of the quadrotor then in third section different control systems proposed and in section four the optimization method presented then in section five compare the results.

2. Quadrotor Mathematical Model

2.1 Working principle

Each of the quadrotor's four rotors (as illustrated in **Figure 1**) has a motor and propeller combination to generate thrust, which lifts the aircraft. Each of these motors is powered independently.

In stationary flights, the front and rear rotors revolve clockwise, while the right and left rotors rotate counterclockwise, resulting in a balanced overall system torque and eliminating gyroscopic and aerodynamic torques.

By changing the speeds of all rotors produces lift force and generates movement; therefore, the decrease o increase in all rotors changes the vertical movement, while the changing in speeds of the second and fourth rotors inversely produces lateral movement with roll rotation.

When the speed of first and third rotors varying conversely, the pitch rotation combined with lateral movement. To generate yaw-motion, the counter-torque must be change between each pair of propellers [8].

2.2 Dynamic model of quadrotor

The quadrotor has two coordinate systems (as shown in **Figure 1**)

I. Body fixed frame (B-frame).
II. Earth fixed frame (E-frame).

The rotation matrix between the E-frame and B-frame presents by three consecutive rotation (yaw, roll and pitch) [1]

$$R=\begin{bmatrix} Cos\varphi Cos\psi & Sin\varphi Sin\theta Cos\psi - Cos\varphi Sin\psi & Cos\varphi Sin\theta Cos\psi + Sin\varphi Sin\psi \\ Cos\theta sin\psi & Sin\varphi Sin\theta Sin\psi + Cos\varphi Cos\psi & Cos\varphi Sin\theta Sin\psi - Sin\varphi Cos\psi \\ -Sin\theta & Sin\varphi Cos\theta & Cos\varphi Cos\theta \end{bmatrix} \quad (1)$$

Because of the following assumptions, the equations of motion are more easily written in the body fixed frame [15]

I. A quadrotor is a rigid symmetrical body.
II. The geometric center and centroid of the quadrotor are in the same position as the origin of the inertial coordinate system.
III. Flight altitude and other parameters have no effect on the quadrotor's resistance and gravity.
IV. Tensions in both directions are proportional to the square of the propeller speed.

The model of the quadrotor system consists of translational and rotational sub-systems, then the equations of motion are generated based on the Newton-Euler formulism and Newton’s second law.

The input control vector U is define as

$$U = [U_1 \quad U_2 \quad U_3 \quad U_4]^T \quad (2)$$

$$\begin{bmatrix} \mathrm{U}_1 \\ \mathrm{U}_2 \\ \mathrm{U}_3 \\ \mathrm{U}_4 \end{bmatrix} = \begin{bmatrix} K_f & K_f & K_f & K_f \\ 0 & -K_f & 0 & K_f \\ K_f & 0 & -K_f & 0 \\ K_M & -K_M & K_M & -K_M \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} \quad (3)$$

Each one of the input vectors can control a motion of the quadrotor. Note that $\mathrm{U}_1 - U_4$ generate the desired altitude, roll angle, pitch angle and the yaw angle.

Then the quadcopter dynamics model can be described as

$$\left.\begin{aligned}
\ddot{x} &= \frac{-U_1}{m}(\sin\varphi\sin\psi + \cos\varphi\cos\psi\sin\theta) \\
\ddot{y} &= \frac{-U_1}{m}(\cos\varphi\sin\psi\sin\theta - \cos\psi\sin\varphi) \\
\ddot{z} &= g - \frac{U_1}{m}(\cos\varphi\cos\theta) \\
\ddot{\varphi} &= \frac{l}{I_{xx}}U_2 - \frac{J_r}{I_{xx}}\dot{\varphi}\Omega_r + \frac{I_{yy}}{I_{xx}}\dot{\psi}\dot{\theta} - \frac{I_{zz}}{I_{xx}}\dot{\theta}\dot{\psi} \\
\ddot{\theta} &= \frac{l}{I_{yy}}U_3 + \frac{J_r}{I_{yy}}\dot{\theta}\Omega_r + \frac{I_{zz}}{I_{yy}}\dot{\varphi}\dot{\psi} - \frac{I_{xx}}{I_{yy}}\dot{\psi}\dot{\varphi} \\
\ddot{\psi} &= \frac{1}{I_{zz}}U_4 + \frac{I_{xx}}{I_{zz}}\dot{\theta}\dot{\varphi} - \frac{I_{yy}}{I_{zz}}\dot{\varphi}\dot{\theta}
\end{aligned}\right\} \quad (4)$$

3. Proposed Controller

Because of its six degrees of freedom and four actuators, a quadrotor is inherently unstable, hence stabilization is critical. In this work the controllers used to control the altitude and attitude, which mean it was used four controllers to control the $\boldsymbol{U_1}$ **to** $\boldsymbol{U_4}$ as shown in **Figure 2**

3.1. PID

PID controllers are employed in a wide variety of controller applications. It is used widely to control the quadrotor system, the PID controller calculates the error, which is the difference between set-point and the feedback, and tries to minimize it. The structure of the PID controller presented in [2]. The general equation of the PID [2]

$$u(t) = K_P e(t) + K_I \int e(t)dt + K_D \frac{de(t)}{dt} \quad (5)$$

where the error

$$\text{e(t)} = desired\ value\ (t) - output\ value\ (t) \quad (6)$$

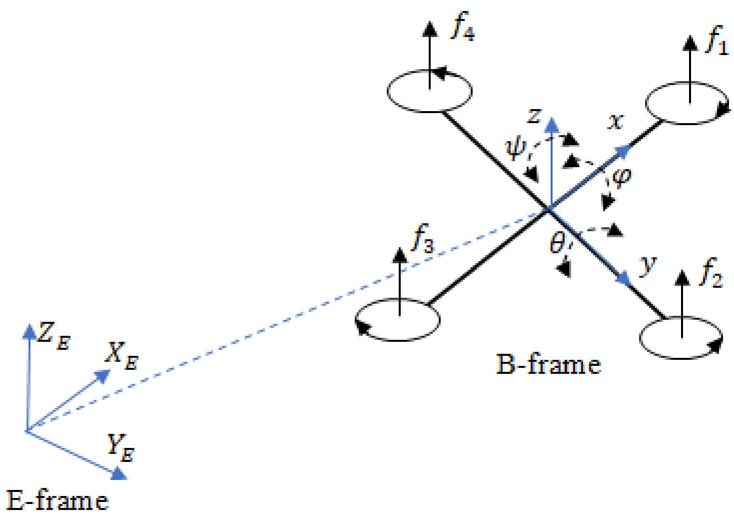


Figure 1: Quadrotor configuration [15].

I. The altitude control equation

$$u_1(t) = K_{Pz}e_z(t) + K_{Iz}\int e_z(t)dt + K_{Dz}\frac{de_z(t)}{dt} \qquad (7)$$

Then the error of the altitude is

$$e_z(t) = z_d(t) - z(t) \qquad (8)$$

where $z_d(t)$ the desired set-point of the altitude, and $z(t)$ the altitude.

II. The roll controller

$$u_2(t) = K_{P\varphi}e_\varphi(t) + K_{I\varphi}\int e_\varphi(t)dt + K_{D\varphi}\frac{de_\varphi(t)}{dt} \qquad (9)$$

where

$$e_\varphi(t) = \varphi_d(t) - \varphi(t) \qquad (10)$$

III. The pitch controller

$$u_3(t) = K_{P\theta}e_\theta(t) + K_{I\theta}\int e_\theta(t)dt + K_{D\theta}\frac{de_\theta(t)}{dt} \qquad (11)$$

where

$$e_\theta(t) = \theta_d(t) - \theta(t) \qquad (12)$$

IV. The yaw controller

$$u_4(t) = K_{P\psi}e_\psi(t) + K_{I\psi}\int e_\psi(t)dt + K_{D\psi}\frac{de_\psi(t)}{dt} \qquad (13)$$

$$e_\psi(t) = \psi_d(t) - \psi(t) \qquad (14)$$

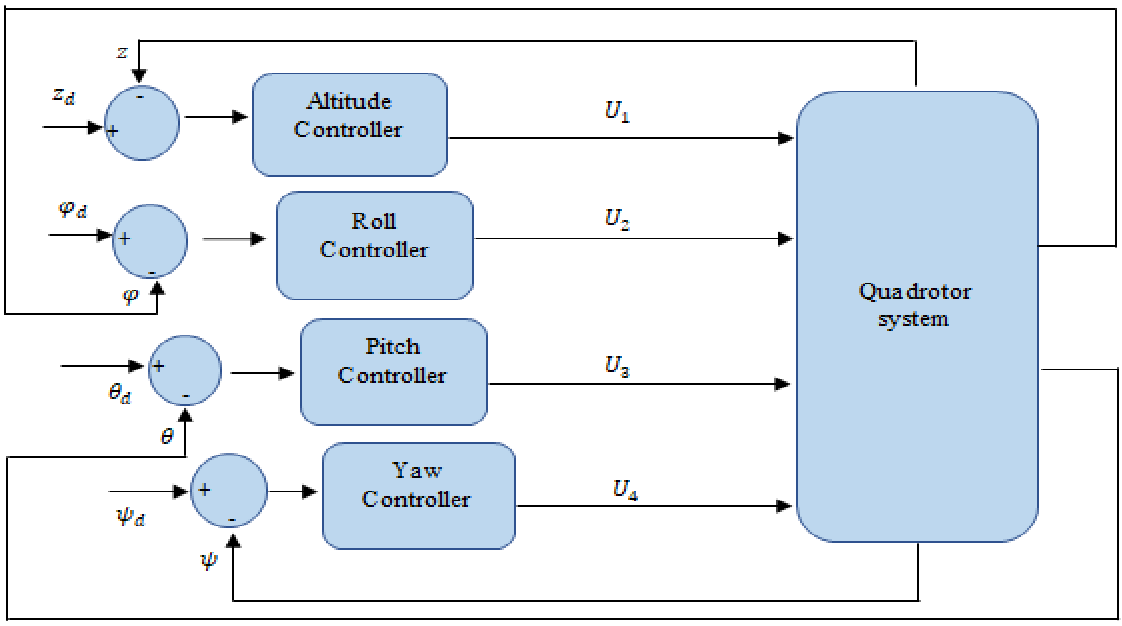


Figure 2: block diagram of altitude and attitude of quadrotor system.

3.2. Fractional Order PID (FOPID) controller

The advent of fractional calculus in recent years has enabled the transfer from classical models and controllers to ones defined by non-integer order differential equations. As a result, fractional-order dynamic models and controllers were developed. It is the general form of the PID with derivatives and integrals in fractional calculus of any order. The FOPID have five parameters (besides the three parameters of the classic PID, there are other two parameters the integral order λ and the derivative order μ; therefore, the general form of the FOPID as it is presented in [16]

$$u(t) = K_P e(t) + K_I D_t^{-\lambda} e(t) + K_D D_t^{\mu} e(t) \qquad (15)$$

where $\mu, \lambda > 0$, Such a controller has more tuning freedom and, as a result, a broader range of parameters that stabilize the plant under control, as well as improved control loop robustness.

Then the equations of FOPID for the altitude and attitude of the quadrotor become

$$u_1(t) = K_{Pz} e_z(t) + K_{Iz} D_t^{-\lambda} e_z(t) + K_{Dz} D_t^{\mu} e_z(t) \qquad (16)$$

$$u_2(t) = K_{P\varphi} e_\varphi(t) + K_{I\varphi} D_t^{-\lambda} e_\varphi(t) + K_{D\varphi} D_t^{\mu} e_\varphi(t) \qquad (17)$$

$$u_3(t) = K_{P\theta} e_\theta(t) + K_{I\theta} D_t^{-\lambda} e_\theta(t) + K_{D\theta} D_t^{\mu} e_\theta(t) \qquad (18)$$

$$u_4(t) = K_{P\psi} e_\psi(t) + K_{I\psi} D_t^{-\lambda} e_\psi(t) + K_{D\psi} D_t^{\mu} e_\psi(t) \qquad (19)$$

where e_z, e_φ, e_θ, and e_ψ are presented in (8), (10), (12), and (14) respectively.

3.3. PID plus second derivative controller (PIDD^2)

It is also known as double derivative PID controller. It is distinguished from typical PID controllers by the addition of a second order derivative term. This contributes to the reduction of vibration error produced by oscillation as well as the reduction of reaction settling time. The equations of the PIDD^2 controller for altitude and attitude of the quadrotor are

$$\boldsymbol{u_1(t) = K_{Pz}e_z(t) + K_{Iz}\int e_z(t)dt + K_{D1z}\frac{de_z(t)}{dt} + K_{D2z}\frac{de_z(t)}{dt}} \quad (20)$$

$$\boldsymbol{u_2(t) = K_{P\varphi}e_\varphi(t) + K_{I\varphi}\int e_\varphi(t)dt + K_{D1\varphi}\frac{de_\varphi(t)}{dt} + K_{D2\varphi}\frac{de_\varphi(t)}{dt}} \quad (21)$$

$$\boldsymbol{u_3(t) = K_{P\theta}e_\theta(t) + K_{I\theta}\int e_\theta(t)dt + K_{D1\theta}\frac{de_\theta(t)}{dt} + K_{D2\theta}\frac{de_\theta(t)}{dt}} \quad (22)$$

$$\boldsymbol{u_4(t) = K_{P\psi}e_\psi(t) + K_{I\psi}\int e_\psi(t)dt + K_{D1\psi}\frac{de_\psi(t)}{dt} + K_{D2\psi}\frac{de_\psi(t)}{dt}} \quad (23)$$

The error signals e_z, e_φ, e_θ, and e_ψ are presented in (8), (10), (12), and (14) respectively.

4. Optimization

The optimization technique is mostly used to identify various ideal decisions or values in order to provide a candidate solution that can effectively address the problem. In general, optimization problems are solved by examining the minimization or maximization of a prospective decision-making procedure. This section presents an optimization method of PID and FOPID controllers' parameters using Salp Swarm Algorithm (SSA).

4.1. Controller gain tuning by SSA

Mirjalili and colleagues[17] presented the Salp swarm algorithm (SSA) as a population-based optimization tool. The SSA's behavior may be demonstrated by computing it using the salp chain in search of optimal food sources (i.e., the target of this swarm is a food source in the search space called F). Individuals salps swarm are classified as leaders or followers in SSA based on their place in the chain. Equation (24) presents the salp-chain, while equation (25) presents the leader position updating and equation (26) shows the updating of the followers

$$X_i = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_d^1 \\ x_1^2 & x_2^2 & \cdots & x_d^2 \\ \vdots & \vdots & \cdots & \vdots \\ x_1^n & x_2^n & \cdots & x_d^n \end{bmatrix} \quad (24)$$

$$x_i^1 = \begin{Bmatrix} y_i + r_1\big((lb_i - lb_i)r_2 + lb_i\big) & r_3 \geq 0 \\ y_i - r_1\big((ub_i - lb_i)r_2 + lb_i\big) & r_3 < 0 \end{Bmatrix} \quad (25)$$

$$x_i^j = \frac{1}{2}\lambda t^2 + \delta_0 t \quad (26)$$

where

x_i^1 : the position of the first salp in the ith dimension

y_i : the food position in the ith dimension.

lb_i and ub_i represent the lower bound and the upper bound of the ith dimension, respectively.

r_1 : Coefficient calculates by $r_1 = 2e^{-\left(\frac{4l}{L}\right)}$

r_2 and r_3 : random numbers between [0-1].

l = current iteration and L = maximum iterations.

x_i^j: Position of followers on the jth salp dimension.

$j > 2\ , \lambda = \frac{\delta_{final}}{\delta_0}$, t: time and $\delta_0 = \frac{x - x_0}{t}$. Assumption$\delta_0 = 0$, then

$$x_i^j = \frac{1}{2}\left(x_i^j - x_i^{j-1}\right) \tag{27}$$

5. Objective Function

The formulation of the objective function is critical in optimization since it is the main parameter used to gauge the success of the optimization approach and determine whether the solution will fit the problem or not.

There are four expression which are functions of an error signal

I. Integral absolute error (IAE).

$$IAE = \int_0^\infty |e(t)|dt \tag{28}$$

II. Integral time absolute error (ITAE).

$$ITAE = \int_0^\infty t|e(t)|dt \tag{29}$$

III. Integral square error (ISE).

$$ISE = \int_0^\infty e^2(t)dt \tag{30}$$

IV. Integral time square error (ITSE).

$$ItSE = \int_0^\infty te^2(t)dt \tag{31}$$

6. Results and Discussion

In this section, the performances of PID, FPID and PIDD2 controllers are tested for control of the UAV system (each controller is used four times to control the inner loop and attitude). The results of the study were achieved using MATLAB/Simulink depending on the parameters of the UAV shown in (table 1).

The SSA algorithm is applied to tune and select the gain parameters of the controllers by using the objective function in section 5 (ITAE in equation 29). Consequently, table 2 lists the setting parameters of SSA with the gains of the controllers. Moreover, (figure 3) shows the step responses for all controllers, and it can be noticed that each controller reaches the steady-state in a different time; whereas, the steady-state error for all controllers equals zero.

Table 3, lists the numerical results of rise time, settling time, overshoot, and integral time absolute error (ITAE). Accordingly, the rise time of the FPID is better than conventional PID; whereas, the PIDD2 has the best rise time. In addition, the settling times of PID and FPID controllers are almost similar, while it is** in PIDD2 controller. Consequently, the overshoot of the FPID is the best compared with the PID and PIDD2 controllers (PIDD2 controller has a maximum overshoot values). Finally, the ITAE for the PIDD2 controller is better than the FPID and PID controllers. Whereas, the PID controller has the maximum values of ITAE. In other words, the performance of PIDD2 is the best in terms of rise time, settling time, and ITAE. Therefore, it can be said that the PIDD2 controller is the fastest.

Table 1: The parameters of quadrotor.

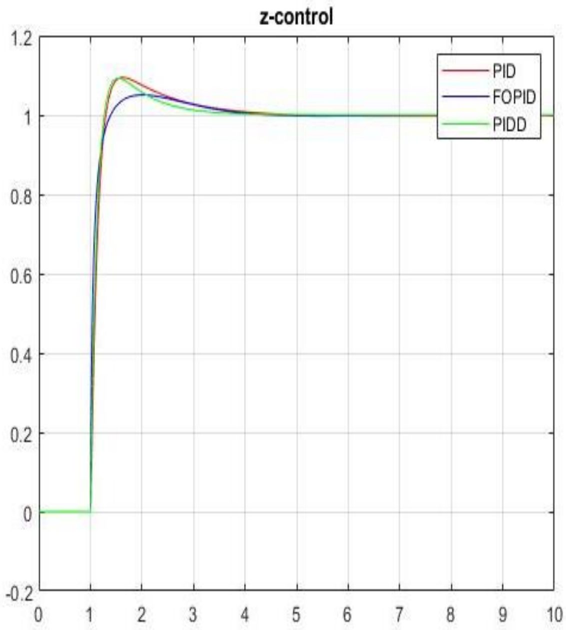
Parameters	Symbol	value	units
Quadrotor moment of inertia around X axis	I_{xx}	$7.5*10^{-3}$	$Kg.m^2$
Quadrotor moment of inertia around Y axis	I_{yy}	$75*10^{-3}$	$Kg.m^2$
Quadrotor moment of inertia around Z axis	I_{zz}	$1.2*10^{-2}$	$Kg.m^2$
Distance to the center of the Quadrotor	l	0.23	m
Mass of the Quadrotor	m	0.65	kg
Gravitational acceleration	g	9.81	m/s^2
Total rotational moment of inertia around the propeller axis	J_r	$6.5*10^{-5}$	$Kg.m^2$
aerodynamic force constant	k_f	$3.13*10^{-5}$	$N.\ s^2$
aerodynamic moments constant	k_m	$7.5*10^{-7}$	$Nm.\ s^2$

Table 2: The setting parameters of SSA with gains' values of the controllers.

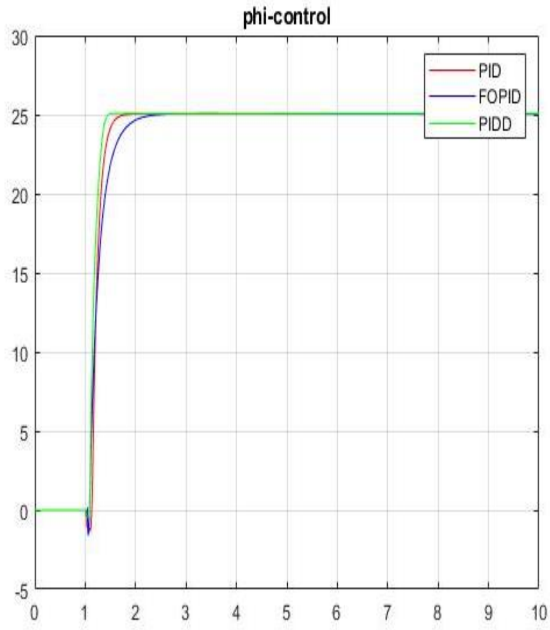
Controller	Population	Iteration		Gains					
				K_P	K_I	K_{D1}	K_{D2}	μ	λ
PID	30	50	z	7.6532	1.5003	7.6502	-	-	-
			Phi	8.3302	0.1135	2.3298	-	-	-
			theta	1.2514	0.0931	0.3120	-	-	-
			Psi	1.1502	0.2389	1.6534	-	-	-
FOPID	10	30	z	11.3203	0.9521	9.3501	-	1.2003	0.5012
			Phi	10.6235	1.9152	3.2015	-	0.9001	0.2398
			theta	9.1230	0.8901	5.3092	-	0.8310	0.4981
			psi	4.0015	2.3692	4.6191	-	0.9810	0.8035
$PIDD^2$	30	30	z	11.5013	0.3025	8.9102	0.5210	-	-
			Phi	4.5928	0.2056	0.5192	0.4136	-	-
			theta	8.6301	1.9410	4.2109	0.4152	-	-
			psi	5.1102	1.7509	2.4158	0.2982	-	-

Table 3: Step Information of controllers with ITAE.

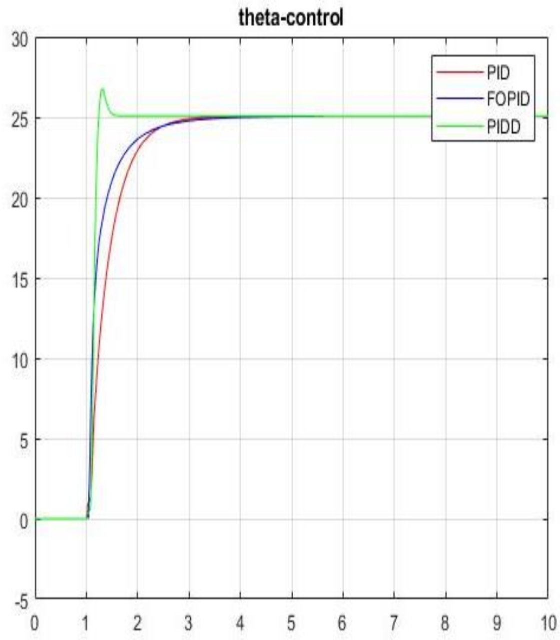
Step Information of PID controller				
	Rise Time	Settling Time	Overshoot	ITAE
z	0.2.003	3.3505	9.6461	0.4696
phi	0.2428	1.5591	0.0184	9.7830
theta	0.8288	2.5754	0.0058	17.0919
psi	0.0524	1.4982	2.4209	3.5930
Step Information of FOPID controller				
z	0.1884	3.2211	5.0832	0.3014
phi	0.4875	2.0184	0.0743	10.7433
theta	0.6515	2.6257	0.0225	12.4308
psi	0.0217	1.0375	1.0888	1.2912
Step Information of $PIDD^2$ controller				
z	0.1834	2.6977	9.2625	0.2845
phi	0.2115	1.3914	0.0554	8.8304
theta	0.1116	1.4280	6.9606	7.6457
psi	0.0323	1.4924	8.1526	1.6035



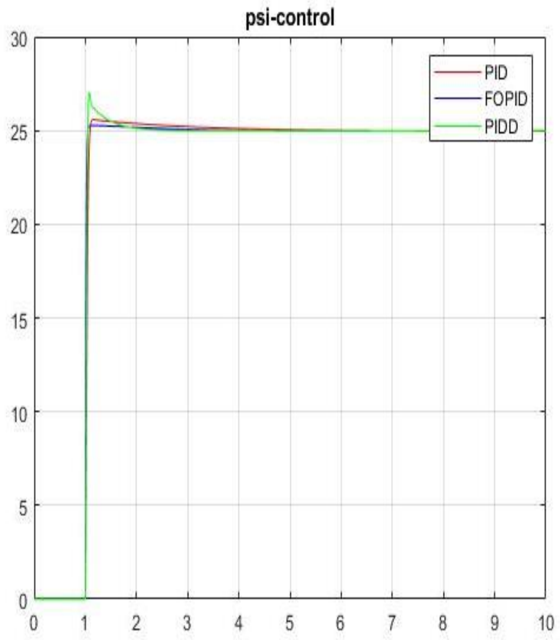
(a) altitude response.



(b) roll response



(c) pitch response.



(d) yaw response.

Figure 3: step responses of the controllers.

7. Related Work Table

Table 4

Study	The Aim	Type of UAV	The controller	Tuned Method
H. Bolandi and colleagues [7] **2012**	To control the attitude of a subsystem	Quadrotor	PID	direct synthesis method
E.Reyes-Valeria and colleagues [6] **2013**	To control the attitude	Quadrotor	LQR Gain Scheduling Control	--
L. Argentim and colleagues[10] **2013**	To compare between different types of controllers	Quadrotor	ITAE, PID and LQR	PID tuned by LQR loop and ITAE tuned by PID
H. Yin and colleagues[12] **2017**	position and attitude tracking control by design a double-loop controller using PID and Sliding mode controller (SMC).	Quadrotor	PD and Sliding mode controller (SMC).	--
B. Kamel and colleagues [9] **2017**	Position and attitude control	Quadrotor	PID	Manual
K. Pan and colleagues[4] **2018**	Track the referred position and attitude	Quadrotor	double closed-loop cascaded LQR	Simultaneous Perturbation Stochastic Approximation (SPSA)
E. Okyere and colleagues [5] **2018**	Position and attitude control	Quadrotor	LQR	MATLAB command by varying Q and R
M. Karahan and colleagues [8] **2019**	Altitude and attitude	Quadrotor	PID	Manual
G. Sheng and colleagues [11] **2019**	Presents the application of fuzzy controller in civil four-rotor UAV	Quadrotor	Classic PID and fuzzy control PID system	Fuzzy
N. Xuan-Mung and colleagues [3] **2019**	Presented a control algorithm for quadrotor's altitude which consists of a combination of nonlinear and linear controllers to control position and attitude	Quadrotor	PID and a new altitude controller which consists of multi-loop controller	--
This paper	Altitude and attitude	Quadrotor	PID, Fractional Order PID, PID plus second derivative controller (PIDD2)	Salp Swarm Algorithm (SSA)

8. Conclusion

In this study, the performance of different controllers are tested for control of a UAV system. The SSA, which was introduced recently, is applied to select the optimal values for the controllers' gains. At first, MATLAB Simulink is used to simulate the dynamic model of the UAV system. Then, SSA is applied to this model based on the integral time absolute error (ITAE), which was selected as the objective function. In spite of the complexity of the UAV system, simulation's results clearly illustrated the good performance of optimized PID, FPID and PIDD^2 controllers. In general, the performance of the PIDD^2 controller is the best. The constraints of the study do not consider the uncertainties and external disturbances. Hence, it can improve the performances of the controllers by adding robust terms that can handle system limitations and constraints.

References

[1] T. Oktay and O. Kose, "Dynamic Modeling and Simulation of Quadrotor for Different Flight Conditions," European Journal of Science and Technology, vol. 15, pp. 132-142, 2019.

[2] N. H. Abbas and A. R. Sami, "Tuning of PID Controllers for Quadcopter System using Hybrid Memory based Gravitational Search Algorithm – Particle Swarm Optimization," International Journal of Computer Applications, vol. 172, pp. 9-18, August 2017.

[3] N. Xuan-Mung and Sung-Kyung Hong, "Improved Altitude Control Algorithm for Quadcopter Unmanned Aerial Vehicles," applied sciences, vol. 9, no. 10, May 2019.

[4] K. Pan, Y. Chen, Z. Wang, H. Wu, and L. Cheng, "Quadrotor Control based on Self-Tuning LQR," in 2018 37th Chinese Control Conference (CCC), Wuhan, China, 2018.

[5] E. Okyere, A. Bousbaine, G. T. Poyi, A. K. Joseph and J. M. Andrade, "LQR controller design for quad-rotor helicopters," The Journal of Engineering, January 2019.

[6] E. Reyes-Valeria, R. Enriquez-Caldera, S. Camacho-Lara and J. Guichard, "LQR control for a quadrotor using unit quaternions: Modeling and simulation," in International Conference on Electronics, Communications and Computing, Cholula, Puebla, Mexico, 2013.

[7] M. R. R. M. H. N. S. M. S. Hossein Bolandi, "Attitude Control of a Quadrotor with Optimized PID Controller," scientific research , vol. 4, pp. 335-342, 2013.

[8] M. Karahan and C. Kasnakoglu, "Modeling and Simulation of Quadrotor UAV Using PID Controller," in International Conference on Electronics, Computers and Artificial Intelligence (ECAI), Pitesti, Romania, 2019.

[9] B. Kamel, B. Yasmina, B. Laredj, I. Benaoumeur and A.-F. Zoubir, "Dynamic modeling, simulation and PID controller of unmanned aerial vehicle UAV," in Seventh International Conference on

Innovative Computing Technology (INTECH), Luton, UK, 2017.

[10] L. M. Argentim, W. C. Rezende, P. E. Santos and R. A. Aguiar, "PID, LQR and LQR-PID on a quadcopter platform," in 2013 International Conference on Informatics, Electronics and Vision (ICIEV), Dhaka, Bangladesh, 2013.

[11] G. Sheng and G. Gao, "Research on the Attitude Control of Civil Quad-Rotor UAV Based on Fuzzy PID Control," in 2019 Chinese Control And Decision Conference (CCDC), Nanchang, China, 2019.

[12] H. Yin, Q. Wang and C. Sun, "Position and attitude tracking control for a quadrotor UAV via double-loop controller," in 2017 29th Chinese Control And Decision Conference (CCDC), Chongqing, China, 2017.

[13] T. Kara and A. Mary, "Adaptive PD-SMC for Nonlinear Robotic Manipulator Tracking Control," Studies in Informatics and Control, vol. 26, pp. 49-58, March 2017.

[14] T. Kara and A. H. Mary, "Robust trajectory tracking control of robotic manipulators based on model-free PID-SMC approach," Journal of engineering research, vol. 6, pp. 170-188, 2018.

[15] A. Noordin, M. A. M. Basri and Z. M. &. I. M. Lazim, "Adaptive PID Controller Using Sliding Mode Control Approaches for Quadrotor UAV Attitude and Position Stabilization," Arabian Journal for Science and Engineering, vol. 46, p. 963–981, 2021.

[16] XiaominTian, Y. Huang and C. Zhang, "The tuning principle of adaptive fuzzy fractional-order PID controller parameters," Procedia Engineering, vol. 7, pp. 251-255, 2010.

[17] A. H. G. S. Z. M. S. S. H. F. S. M. M. Seyedali Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," Advances in Engineering Software, vol. 114, pp. 163-191, 2017.