Performance Comparison of the ANFIS based Quad-Copter Controller Algorithms

Namal Rathnayake
School of Systems Engineering
Kochi University of Technology
Kami City, Kochi, Japan
namalhappy@gmail.com
246007h@gs.kochi-tech.ac.jp

Tuan Linh Dang
School of Information and Communications Technology
Hanoi University of Science and Technology
Hanoi, Vietnam
linhdt@soict.hust.edu.vn
linh.dangtuan@hust.edu.vn

Yukinobu Hoshino
School of Systems Engineering
Kochi University of Technology
Kami City, Kochi, Japan
hoshino.yukinobu@kochi-tech.ac.jp

***Abstract*—Performing an accurate and smooth trajectory of a quad-copter is a crucial aspect in autonomous controls due to its non-linearity and under-actuated characteristic. Adaptive Neuro-Fuzzy Inference System (ANFIS) is well-known for non-linear controls. This paper focuses on comparing the performance of ANFIS based quad-copter systems to identify the best optimization algorithm. Two famous algorithms called Genetic Algorithms (GA) and Particle Swarm Optimization (PSO) was used as the optimization algorithms and to tune the gains of the Fuzzy Inference Systems (FIS). The analysis was performed using two different simulations namely, altitude control and trajectory navigation. The final results were compared between traditional PID, conventional ANFIS, GA-ANFIS and PSO-ANFIS. PSO-ANFIS obtained the highest performance in our experiments.**

***Index Terms*—Quad-Copter control, Adaptive Neuro Fuzzy Inference System (ANFIS), Genetic Algorithms (GA), MATLAB, Particle Swarm Optimization (PSO)**

I. Introduction

Quad-copter controller designing is a trending topic of today's world researchers, because, the possibilities and capabilities of the quadcopter provide high expectations. The quad-copters are usually used in the fields of agriculture [1], [2], areal transportation [3], [4], mapping and exploration, fire and rescue [5], [6] and photography [7], [8].

Quad-copter has four motors, while the degrees of freedom are six. Hence, it is an underactuated mechanical system. Due to the under-actuation of the quad-copter, designing a better control algorithm is considerably an extensive challenge. Recently there have been several quad-copter control algorithms which provide better performance in stabilization when compared with the early stage controllers. PID control is one of the most used algorithms in quad-copter controls [9]–[11]. Most of the researchers tend to use the PID control algorithm as the initial step of developing the novel algorithms [12]–[14]. some of these can be listed as follows. Neural network (NN) based controllers [15], Neural-PID based controllers [16], sliding mode controllers [16], Linear quadratic controllers [17] and ANFIS based controllers [18], [19].

Among these control algorithms, it is a proven fact that the ANFIS based controllers provide better efficiency and stability according to the following literature. The authors in [20] have developed a quad-copter using ANFIS algorithm to control the attitude and altitude. The authors have compared the results between ANFIS-PD, FUZZY-PD and PD controllers. The results show that the ANFIS based controller has better accuracy in attitude and altitude controlling over the other two algorithms. In [21], a Permanent Magnet Brush-less DC motor drive system is implemented using ANFIS and PI controllers. The results show that the ANFIS provides better results in current and torque controlling over the fixed PI controller. An altitude tracking system for a drone is implemented in [22]. They have used hybrid PD-ANFIS based algorithm for controlling. The results were obtained for classical PID, P-D, ANFIS and Intelligent PD-ANFIS controllers. Though all the considered controllers were able to stabilize the drone, the PD-ANFIS gives faster stability as in the results section of the paper.

The authors in [23] have presented an algorithm for drone control using ANFIS and Improved Particle Swarm Optimization algorithm (ANFIS-IPSO). The results were compared between PID, ANFIS and proposed algorithm. The trajectory tracking error has almost come to a zero when using ANFIS-IPSO algorithm. In [18], a trajectory control system for a quad-rotor is designed using PSO-ANFIS algorithm. Two simulation tests have been carried out for performance analysis such as trajectory tracking and mass loading and unloading. They compare the results with PID, ANFIS and PSO-ANFIS algorithms and the results prove that PSO-ANFIS provides better stability in both simulations. Hence, it is possible to do a performance analysis among state-of-the-art ANFIS based algorithms.

The main goal of this paper is to investigate the performance of the GA-ANFIS and PSO-ANFIS in quad-copter control.

The rest of the paper is organized as follows. Section II gives a brief introduction of the related theories such as quadcopter model dynamics, PID controllers, ANFIS, GA and PSO. Then the methodology of the controller implementation is presented in section III. As section IV, the results were analysed and finally, conclusion and future works are presented as section V.

978-1-6654-4407-1/21/$31.00

II. Related Theories

A. Quad-copter model dynamics

It is a necessary fact to understand the basic dynamics of the quad-copter designing before implementing a control algorithm. As mentioned in the above section, quad-copter is a system which is under-actuated. Because there are six degrees of freedoms in the quad-copter, but it has four rotors to control the movements. The six degrees of freedoms can be divided into two as, translation movements (along the x-axis, y-axis and z-axis) and rotational angles (φ - roll, θ - pitch and ψ - yaw). This section is used to give a brief introduction about the quad-copter model dynamics.

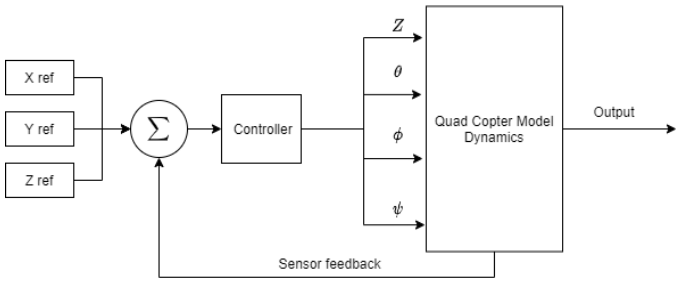

Fig. 1. Quad-Copter Controller

Since there are two types of freedoms, the kinematics can be divided into two sections as well. As referenced in [18], the translation kinematics (1) and rotational kinematics (2) can be presented as the following equations.

$$\dot{\xi} = R.V_b \tag{1}$$

$$\dot{\eta} = R_r.\mho_b \tag{2}$$

Here absolute linear velocity vector is presented as $\dot{\xi}$ and it can be expressed as $\begin{bmatrix}\dot{x} & \dot{y} & \dot{z}\end{bmatrix}$. Linear velocity is expressed using V_b and it can be presented as $\begin{bmatrix}U & V & W\end{bmatrix}^T \in \mathbb{R}^3$. R can be given as:

$$R_r = \begin{bmatrix} 1 & s_\varphi t_\theta & c_\varphi t_\theta \\ 0 & c_\varphi & -s_\varphi \\ 0 & \frac{s_\varphi}{c_\theta} & \frac{c_\varphi}{c_\theta} \end{bmatrix} \tag{3}$$

where euler angle derivatives can be presented as $t_\theta = \tan\theta$, $\dot{\eta} = \begin{bmatrix}\dot{\varphi} & \dot{\theta} & \dot{\psi}\end{bmatrix}^T \in \mathbb{R}^T$

As in [24], the rotor forces can be expressed as in (4) and the model dynamics can be represented using (5).

$$f_i = b.{\omega_i}^2 \tag{4}$$

$$\begin{cases} \dot{x} = u \\ \dot{y} = v \\ \dot{z} = w \\ \dot{u} = (c_\varphi s_\theta c_\psi + s_\varphi s_\psi)\frac{T}{m} \\ \dot{v} = (c_\varphi s_\theta c_\psi - s_\varphi c_\psi)\frac{T}{m} \\ \dot{w} = (c_\varphi c_\theta)\frac{T}{m} - g \\ \dot{\varphi} = p + s_\varphi t_\theta q + c_\varphi t_\theta r \\ \dot{\theta} = c_\varphi q - s_\varphi r \\ \dot{\psi} = \frac{s_\varphi}{c_\theta} q + \frac{c_\varphi}{c_\theta} r \\ \dot{p} = \frac{\tau_x}{j_x} + \frac{j_y - j_z}{j_x} qr \\ \dot{p} = \frac{\tau_y}{j_y} + \frac{j_z - j_x}{j_y} pr \\ \dot{r} = \frac{\tau_z}{j_z} + \frac{j_x - j_y}{j_z} pq \end{cases} \tag{5}$$

In (4), rotor thrust is denoted as b and angular velocities are ω. The inertia matrix of the quad copter is denoted as j and j_x, j_y and j_z are the moments of inertia in (5).

B. Proportional Integral and Derivative (PID) control

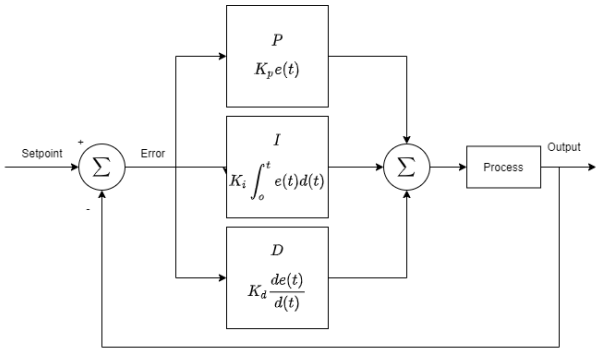

Fig. 2. Classical PID Controller

PID control is the most commonly used control technology in control engineering. This section aims to give a brief introduction about the PID controllers. It is said that PID controllers are the workhorse of the modern process control systems [25]. PID controllers are made with three components such as proportional, integral and derivative. The basic structure of the PID controller in a system is shown in Fig. 2. As shown in the figure, the output is controlled by three parameters (P, I, D). Generally, PID controllers are closed-loop controllers. Therefore, the error $e(t)$ is calculated using the feedback path and the setpoint to control the system. In the figure, K_p, K_i and K_d are the gains of proportional, integral and derivative respectively.

The tuning of theses gains is challenging. Hence, many researchers have proposed several types of PID tuning methods. Some of them are; trial and error Method [26], Zeigler-Nichols Method [27], Cohen-Coon method [28], Tyreus-Luyben method [29] and autotune method. Moreover, there are some software-based tuning methods such as MATLAB PID tuning [30].

C. ANFIS

In 1978, L.A. Zadeh [31] introduced Fuzzy Logic Controllers (FLC). Since then, it has evolved to a significant level. The recent evolution of FLC is ANFIS. ANFIS is based on two well-known algorithms such as FLC and the ANN. Jang in 1993 [32] proposed ANFIS as a machine learning algorithm. In this section, a brief introduction about ANFIS is presented.

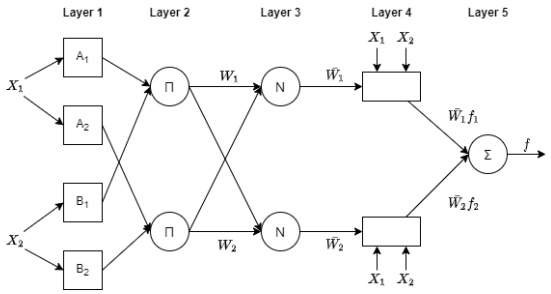

Fig. 3. ANFIS Architecture

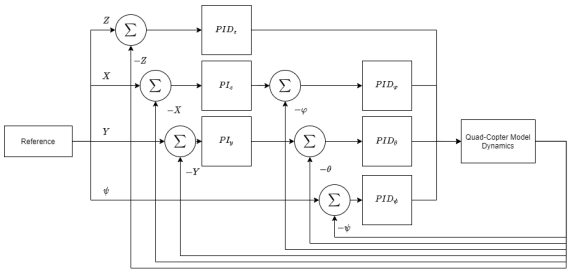

Fig. 4. PID control of Quad-Copter

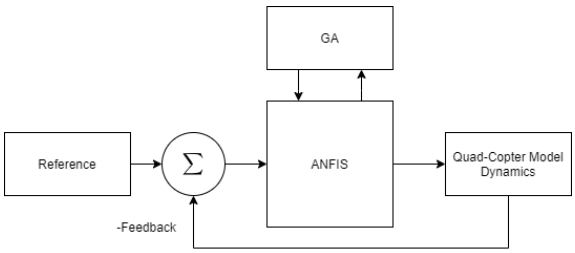

Fig. 5. GA-ANFIS control of Quad-Copter

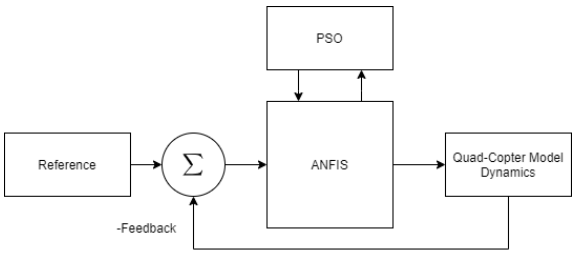

Fig. 6. PSO-ANFIS control of Quad-Copter

Fig. 3 shows the general structure of the ANFIS algorithm. ANFIS possess all the advantages of the ANN and the FLC [33]. Generally, ANFIS is a five-layer structure as shown the figure. Each layer is explained in brief below.

- Layer 1: The first layer is called the fuzzification layer. Here, the memberships are generated for each inputs using the standard membership functions as in (6).

$$O_{1,i} = \begin{cases} \mu_{Ai}(X_1) \\ \mu_{Bi}(X_2) \end{cases} \tag{6}$$

where, linguistic labels are denoted as A_i and B_i while node function is denoted as i. There are three commonly used membership functions such as triangular (7), trapezoidal (8) and gaussian (9).

$$\mu_{Ai} = max(min(\frac{x - a_i}{b_i - a_i}, \frac{c_i - x}{c_i - b_i}), 0), i = 1, 2 \tag{7}$$

$$\mu_{Ai} = max(min(\frac{x - a_i}{b_i - a_i}, 1, \frac{d_i - x}{d_i - c_i}), 0), i = 1, 2 \tag{8}$$

$$\mu_{Ai} = exp(-\frac{(x - c_i)^2}{\sigma_1^2}), i = 1, 2 \tag{9}$$

Here, a_i, b_i, c_i, d_i and σ_i are premise parameters.

- Layer 2: This later is called the rule layer. As in (10), the firing strength w of the each rule is generated by multiplying $\mu_{Ai}(x_1)$ and $\mu_{Bi}(x_2)$.

$$O_{2,j} = w_j = \mu_{Ai}(x_1) \times \mu_{Bi}(x_2) \tag{10}$$

where j is the number of the rule.

- Layer 3: This is the normalization layer. The main objective of this later is to generate the third output by normalization by firing strength of the each corresponding rule. Normalization can be expressed as in (11).

$$O_{3,j} = \bar{w_j} = \frac{w_j}{\sum w_i}, j = 1, 2, ..., n \tag{11}$$

- Layer 4: This layer is responsible for the defuzzification. Consequent parameters (p_j, q_j, r_j) are used here as in (12) to do the defuzzification of the previously generated normalized firing strengths $\bar{w_j}$.

$$O_{4,j} = \bar{w_j} f_j = \bar{w_j}(p_j x_1 + q_j x_2 + r_j) \tag{12}$$

- Layer 5: This is the output layer. Sum of all incoming inputs gives the overall output as in (13).

$$O_{5,1} = \sum_{j=1}^{n} \bar{w_j} f_j \tag{13}$$

D. Fitness Function

The relative importance of a design can be expressed using a fitness function. Determining the best fitness function for a certain system is a assuring way to collect better results. The most common way to describe a fitness function is using a cost function as shown in (14) [34].

$$F_i = (1 + \varepsilon) f_{max} - f_i \tag{14}$$

where, the cost function is denoted as f_i for the ith design, f_{max} is max cost and ε is used to eliminate the difficulty of F_i being zero. Considering the quad copter control, Integral of Time multiplied by Absolute Error (ITAE) is most used method in determining a fitness function. The adjusting time and the overshoot is indicated by ITAE. These characteristics reflects the accuracy and the rapidity of the control system. It is scientifically proven that (15) gives the performance index of a quad copter control [35].

$$J_{ITAE} = \int_0^\infty t.\,|e(t)|\,dt \tag{15}$$

E. Genetic Algorithms (GA)

The inspirations of Genetic Algorithms (GA) are natural selections and biological process. There are several versions of GA in the literature. This section gives a brief introduction about GA and its construction.

GA is an optimization algorithm which tries to find the optimum solution in a solution space by iterations. First, the search for the best solution starts with a randomly generated population in the solution space. In order to continue with the iterations, GA proceed with three operators namely, selection, crossover and mutation [36].

The principle of "Survival of the Fittest" imitates the first operation "Selection". Considering the current design set which is also called the population, this process is predisposition towards the better fit members. In (16), this process is expressed in mathematically [34].

$$P_i = \frac{F_i}{Q}; Q = \sum_{j=1}^{N_p} F_j \tag{16}$$

Here, F_i is the fitness value. N_p is the number of members in the population and P_i is the probability of the selection.

The second operator "Crossover" is about the mating in biological population. There are several crossover methods in completing this operation such as single-point crossover, two-point and k-point crossover, uniform crossover. Crossover helps to sustain the good solutions and remove the bad solutions and move on with the new population to the next generation. Diversity in characteristics of the population is done by the last operator "Mutation". Mutation helps to escape from the local minima and propagate to the global minima.

Generally, GA encodes the design into bits of ones and zeros. Hence, GA is well known for discrete designs because of this behaviour of the algorithm.

F. Particle Swarm Optimization (PSO)

This section is used to describe the PSO algorithm in brief. PSO is firstly introduced in 1997 [37] by the inspirations of birds flocking. PSO is also an optimization algorithm which tries to move in the solution space to find the best solution possible called global minima. PSO also operates in three steps namely, positions and velocity generations for each particle, updating the velocities and updating the positions. PSO also uses iterations to complete the search. The first step of the PSO is to generate random population with random positions and velocities as in (17) and (18).

TABLE I
MODEL PARAMETERS OF THE QUAD COPTER

Description	**Value**
Gravitational Acceleration (g)	$9.81ms^{-2}$
Quad copter mass (m)	$0.65\ kg$
Distance from center to motor (l)	$0.23m$
About x axis moment of inertia (jx)	$0.0075kgm^2$
About y axis moment of inertia (jy)	$0.0075kgm^2$
About z axis moment of inertia (jz)	$0.013kgm^2$
Force constant of propellers (k)	$0.0000313Ns^2$
Torque constant of propellers (d)	$0.00000075Ns^2$

$$\boldsymbol{x}_0^i = \boldsymbol{x}_{min} + rand(\boldsymbol{x}_{max} - \boldsymbol{x}_{min}) \tag{17}$$

$$\boldsymbol{v}_0^i = \frac{\boldsymbol{x}_{min} + rand(\boldsymbol{x}_{max} - \boldsymbol{x}_{min})}{\Delta t} = \frac{position}{time} \tag{18}$$

where $\boldsymbol{x_{min}}$ and $\boldsymbol{x_{max}}$ are the lower and upper bounds respectively.

At each iteration, the velocities and positions are updated as in the following equations (19)(20).

$$\boldsymbol{v}_{k+1}^i = w\boldsymbol{v}_k^i + c_1 rand\frac{p^i - \boldsymbol{x}_k^i}{\Delta t} + c_2 rand\frac{p_k^g - \boldsymbol{x}_k^i}{\Delta t} \tag{19}$$

$$\boldsymbol{x}_{k+1}^i = \boldsymbol{x}_k^i + \boldsymbol{v}_{k+1}^i \Delta t \tag{20}$$

where, $\boldsymbol{v}_{\boldsymbol{k}+1}^{\boldsymbol{i}}$ is the velocity of the particle i at time $\boldsymbol{k}+1$, $\boldsymbol{w}$ is the inertia factor, $\boldsymbol{w}\boldsymbol{v}_{\boldsymbol{k}}^{\boldsymbol{i}}$ is the current motion, $\boldsymbol{c_1}$ is the self confidence, $\boldsymbol{c_1 rand}\frac{\boldsymbol{p^i - x_k^i}}{\boldsymbol{\Delta t}}$ is the particle memory influence, c_2 is the swarm confidence and $\boldsymbol{c_2 rand}\frac{\boldsymbol{p_k^g - x_k^i}}{\boldsymbol{\Delta t}}$ is the swarm influence. $\boldsymbol{p}^i$ is the pbest position of the particle i. Each particle keeps swam in the coordinates solution space, which is associated with the best solution (fitness) that has achieved. This value is called the personal best(pbest). Another $\boldsymbol{p^g}$ is the gbest position. The best value that is taken by the PSO, is the best value obtained by all particles. This value is called the global best (gbest).

III. METHODOLOGY

The above-introduced control algorithms were implemented using SIMULINK environment in MATLAB. The quad-copter dynamics are shown in Table I.

Quad-copter is a system which has six controllers for each degree of freedoms namely, pitch (φ), roll (θ) and yaw (ψ) Euler angles and X, Y and Z position coordinates. Hence, the initial system is designed to use six PID controllers ($PID_\varphi, PID_\theta, PID_\psi, PI_X, PI_Y, PID_Z$). The overall view of the PID based quad-copter system is presented in Fig. 4. Note that, X and Y PI controllers are placed in cascaded with θ and ψ PID controllers respectively.

The PID tuning was done using the Zeigler-Nichols method [27]. Each PID controller was tuned starting with the altitude controller (Z) separately. The Table II shows all tuned parameters used in the PID controllers.

TABLE II
ALL PID PARAMETERS (ZEIGLER-NICHOLS)

	X	Y	Z	φ	θ	ψ
K_p	0.2	0.2	0.6	0.51	0.51	0.105
K_i	0.1	0.1	0.35	0.81	0.81	0.005
K_d	0.0	0.0	0.43	0.81	0.81	0.005

TABLE III
GA PARAMETERS

Iterations	100
Population Size	40
Crossover Percentage	50% Uniform crossover
Mutation Percentage	0.5%
Number of Mutants	20
Gamma (γ)	0.2
Mutation Rate (μ)	0.1
Selection Pressure (β)	8

Using the above tuned PID parameters, the output response was collected against the input and recorded for the ANFIS algorithm training.

The recorded data were divided into two segments as training and testing at the ratio of 70% and 30%. $Genfis2$ function in MATLAB is used as the initial membership function generation in FIS. The training was conducted offline and then the error was observed by the testing phase.

GA and PSO were used to optimize the FISs which were initially generated using ANFIS algorithm. These two optimization algorithms were used in the optimal parameter settings as mentioned in the literature [38]. The parameter setting of the GA and PSO is shown the Table III and IV.

Designing of the GA-ANFIS and PSO-AFNIS controllers were performed as in the following pseudo-codes. Algorithm 1 shows the main steps in tuning the FIS parameters using GA. As introduced before, the tree main operations were implemented to search for the best solution in the solution space. The overall representation of the GA-ANFIS algorithm in the quad-copter system is shown in Fig. 5.

The construction of PSO-ANFIS is presented in Algorithm 2. First, the initial FIS is generated using ANFIS algorithm and then PSO is used to tune the FIS parameters furthermore. The overall representation of the PSO-ANFIS algorithm in the quad-copter system is shown in Fig. 6.

IV. RESULTS AND DISCUSSION

The experiments were conducted using two different simulations such as altitude response measurement and way-point navigator to find out the effectiveness of the corresponding

TABLE IV
PSO PARAMETERS

Iterations	100
Population Size	40
Inertia Weight (w)	0.5
Inertia Weight Damping ratio	0.99
Personal Learning Coeffition (c_1)	1.5
Global Learning Coeffition (c_2)	1.5

Algorithm 1: GA-ANFIS optimization

Result: Optimized GA-FIS
Initial FIS;
begin
  Data acquisition from PID controllers;
  Random Partitioning of data as training and testing;
  Initial Fuzzy Inference System (FIS) generation using the training set;
end
GA tuning;
begin
  Selection;
  Crossover;
  Mutation;
end
Store the best cost and best solution;
Update the FIS;

Algorithm 2: PSO-ANFIS optimization

Result: Optimized PSO-FIS
Initial FIS;
begin
  Data acquisition from PID controllers;
  Random Partitioning of data as training and testing;
  Initial Fuzzy Inference System (FIS) generation using the training set;
end
PSO tuning;
begin
  Generate Random Population;
  Velocity update;
  Position update;
end
Store the best cost and best solution;
Update the FIS;

TABLE V
RMSE OF EACH ANFIS CONTROLLERS

Error	**Algorithm**	**Pitch**	Roll$\times 10^{-5}$	**Yaw**$\times 10^{-8}$	**Z**
RMSE	**ANFIS**	0.000124	2.889	1.755	0.00434
	GA-ANFIS	0.000122	2.784	1.594	0.00337
	PSO-ANFIS	0.000112	2.782	1.585	0.00198

TABLE VI
TRAINING TIME OF EACH ANFIS CONTROLLERS

	Algorithm	**Pitch**	**Roll**	**Yaw**	**Z**
Time(s)	**GA-ANFIS**	705.32	696.32	705.90	697.97
	PSO-ANFIS	588.03	593.22	596.57	572.88

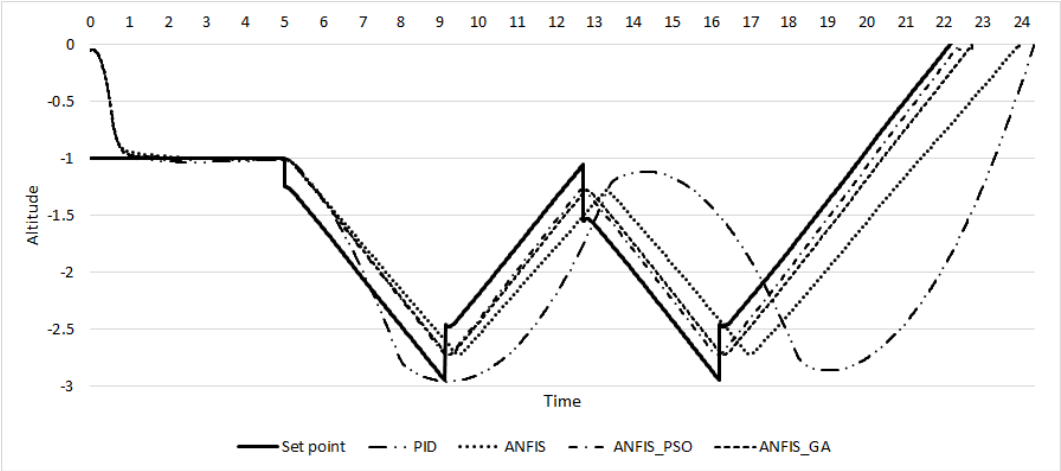

Fig. 7. Altitude Response

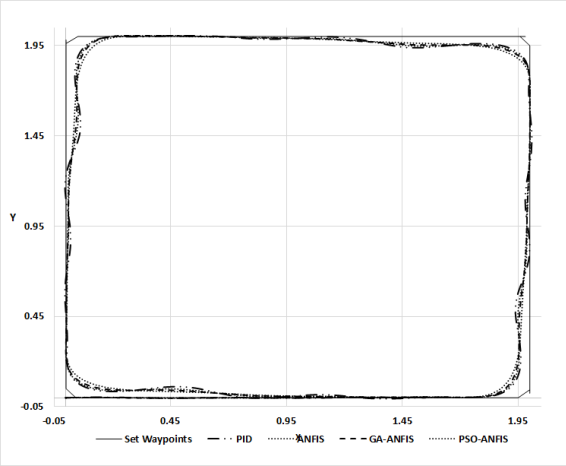

Fig. 8. Way-point Follower

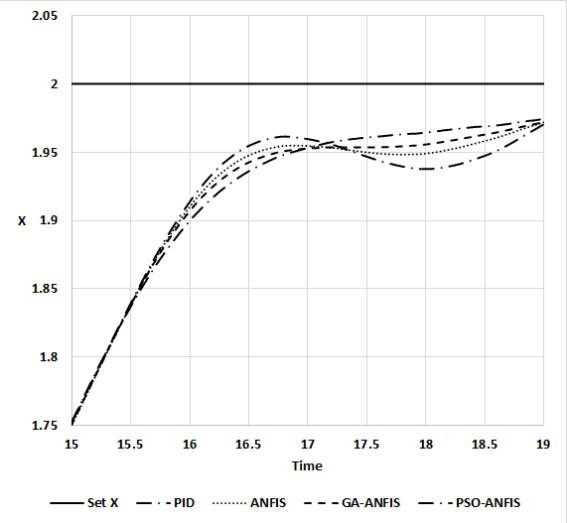

Fig. 9. Change in X coordinates (I)

control algorithms. The results were obtained for PID, ANFIS, GA-ANFIS and PSO-ANFIS.

As the first simulation test, the altitude response was measured. A MATLAB simulation was designed to change the altitude for 22 seconds. The set coordinates were given in the z-axis which is the altitude in this system. The results can be presented in the Fig. 7.

As shown in this figure, PID, ANFIS, PSO-ANFIS and GA-ANFIS performance of obtaining the desired setpoints are shown. When compare the PID response with the other three ANFIS algorithms, the ANFIS based systems show a smoother and more accurate response. It is also clear to mention that the response delay is much larger in the PID controller than ANFIS controllers.

Among the ANFIS controllers, such as ANFIS, GA-ANFIS and PSO-ANFIS, the best results were given by the PSO-ANFIS controller.

The next simulation was conducted to obtain the performance in X and Y axis navigation. In this simulation, five set points were defined as $(0, 0, -1), (2, 0, -1), (2, 2, -1), (0, 2, -1), (0, 0, -1)$. Here, the first second and third coordinates are X, Y and Z respectively. The altitude kept in a constant level at $1m$ and X and Y coordinates were changed accordingly. The results are presented using Figs. 8, 9, 10, 11, and 12.

Fig. 8 shows the total way-point navigation of the drone in X and Y plane. Fig. 9 and Fig. 10 are the closeup views of two instances in X coordinate variations. As in these figures, it is clear that the PSO-ANFIS gives better performance in less overshoot and faster stability. Y coordinate variations are shown in the Fig. 11 and Fig. 12. Here, the results are similar to the X coordinate variations. The numerical data for steady state errors for the PID, ANFIS, GA-ANFIS and PSO-ANFIS are 0.00861, 0.00212, 0.00106, and 0.00009 respectively. The Rising times are $9s$, $8.73s$, $8.69s$ and $9.12s$ accordingly.

The results in Figs. 7, 8, 9, 10, 11, and 12 proves the superi-

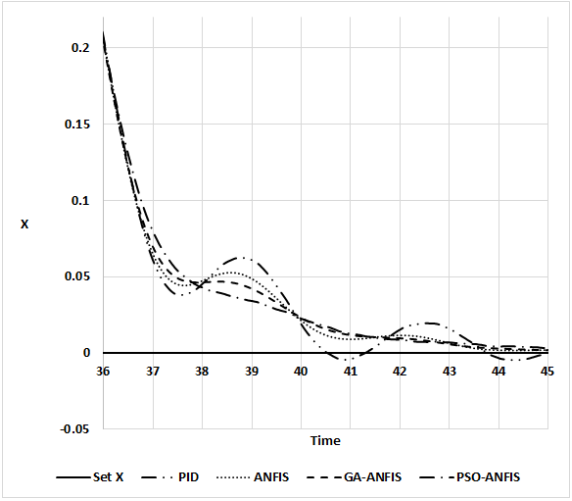

Fig. 10. Change in X coordinates (II)

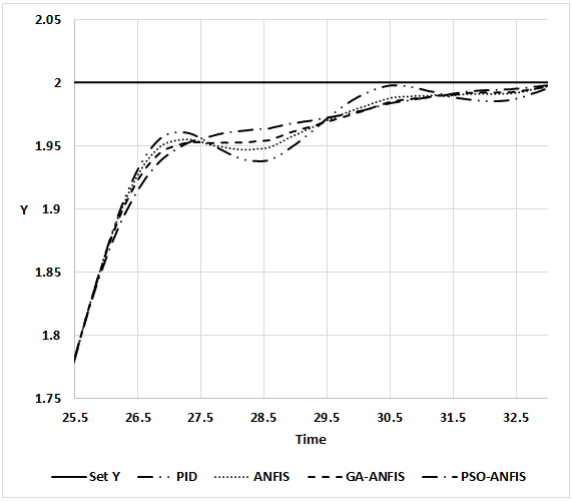

Fig. 11. Change in Y coordinates (I)

ority of PSO-ANFIS and GA-ANFIS over Conventional ANFIS and PID controllers. Both the nature inspired algorithms (GA-ANFIS and PSO-ANFIS) shows better performance than the ANFIS and traditional PID controllers.

Moreover, the Table V shows the Root Mean Squared Error (RMSE) for each controller. It can be observed that PSO-ANFIS and GA-ANFIS show very similar results. However, PSO has less error and faster stability than GA. This may be caused due to the following differences in the considered algorithms [39].

The inbuilt guidance mechanism in PSO causes faster convergence. But in GA, there is no dedicated guidance mechanism. Only when the particles participate in the crossover, the better solutions converge. PSO handles two populations at a time, namely, $pbest$ (personal best) and $current\ positions$. This key feature helps the PSO to re-navigate in the solution space if the solution is moving towards an unwanted path. GA does not handle two populations. Hence this becomes a huge

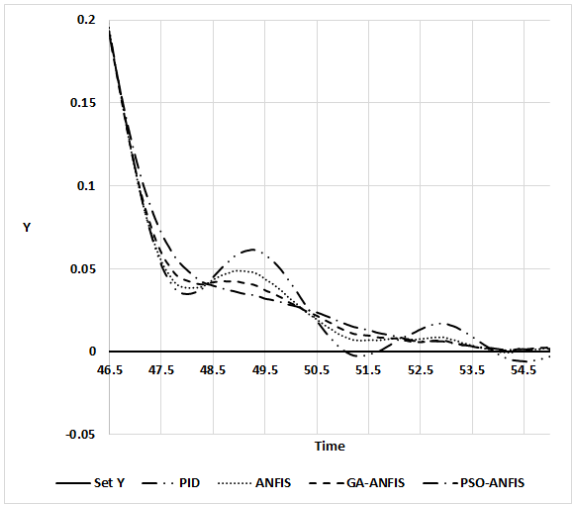

Fig. 12. Change in Y coordinates (II)

advantage in solution exploration. The algorithm construction of the PSO is purely mathematical while the GA has non-mathematical operations such as crossover operation. This gives the benefit of less time consumption in PSO than GA and it can be seen as in the Table VI.

V. Conclusion and Future Works

In this paper, performance analysis is conducted to compare the results among the evolution of ANFIS controllers with optimization algorithms such as GA and PSO. A quad-copter is used here for the control application. We have also used the traditional PID controller for initial tuning of all other algorithms and then individually optimized the rest. The experiments were conducted as simulations. Two simulations such as altitude test and way-point follower were used to obtain the response in X, Y and Z axis navigation. As shown in the results section in this paper, the PSO-ANFIS algorithm outperforms all the other control algorithms. Both of the simulations show that the PSO-ANFIS is better in quad-copter applications because it has lesser overshoot and stabilizes faster than the other algorithms used here. As for the future objective, the implementation in real-time and performance analysis can be indicated.

References

[1] U. R. Mogili and B. Deepak, "Review on application of drone systems in precision agriculture," *Procedia computer science*, vol. 133, pp. 502–509, 2018.

[2] G. DEVI, N. Sowmiya, K. Yasoda, K. Muthulakshmi, and K. BALASUBRAMANIAN, "Review on application of drones for crop health monitoring and spraying pesticides and fertilizer," *J. Crit. Rev*, vol. 7, pp. 667–672, 2020.

[3] N. Cohen, "Aerial transport system," June 11 2009. US Patent App. 12/300,126.

[4] F. Yakushiji, K. Yakushiji, M. Murata, N. Hiroi, K. Takeda, and H. Fujita, "The quality of blood is not affected by drone transport: An evidential study of the unmanned aerial vehicle conveyance of transfusion material in japan," *Drones*, vol. 4, no. 1, p. 4, 2020.

[5] B. Mishra, D. Garg, P. Narang, and V. Mishra, "Drone-surveillance for search and rescue in natural disaster," *Computer Communications*, vol. 156, pp. 1–10, 2020.

[6] R. Tariq, M. Rahim, N. Aslam, N. Bawany, and U. Faseeha, "Dronaid: A smart human detection drone for rescue," in *2018 15th International Conference on Smart Cities: Improving Quality of Life Using ICT & IoT (HONET-ICT)*, pp. 33–37, IEEE, 2018.

[7] H. Schmidt, "From a birdâs eye perspective: Aerial drone photography and political protest. a case study of the bulgarian# resign movement 2013," *Digital Icons: Studies in Russian, Eurasian and Central European New Media*, vol. 13, pp. 1–27, 2015.

[8] A. Puttock, A. Cunliffe, K. Anderson, and R. E. Brazier, "Aerial photography collected with a multirotor drone reveals impact of eurasian beaver reintroduction on ecosystem structure," *Journal of Unmanned Vehicle Systems*, vol. 3, no. 3, pp. 123–130, 2015.

[9] J. O. Escobedo-Alva, E. C. Garcia-Estrada, L. A. Paramo-Carranza, J. A. Meda-Campana, and R. Tapia-Herrera, "Theoretical application of a hybrid observer on altitude tracking of quadrotor losing gps signal," *IEEE Access*, vol. 6, pp. 76900–76908, 2018.

[10] C. Aguilar-Ibanez and M. S. Suarez-Castanon, "A trajectory planning based controller to regulate an uncertain 3d overhead crane system," *International Journal of Applied Mathematics and Computer Science*, vol. 29, no. 4, 2019.

[11] J. R. García-Sánchez, S. Tavera-Mosqueda, R. Silva-Ortigoza, V. M. Hernández-Guzmán, J. Sandoval-Gutiérrez, M. Marcelino-Aranda, H. Taud, and M. Marciano-Melchor, "Robust switched tracking control for wheeled mobile robots considering the actuators and drivers," *Sensors*, vol. 18, no. 12, p. 4316, 2018.

[12] D. I. Martinez, J. J. De Rubio, T. M. Vargas, V. Garcia, G. Ochoa, R. Balcazar, D. R. Cruz, A. Aguilar, J. F. Novoa, and C. Aguilar-Ibanez, "Stabilization of robots with a regulator containing the sigmoid mapping," *IEEE Access*, vol. 8, pp. 89479–89488, 2020.

[13] D. I. Martinez, J. d. J. Rubio, V. Garcia, T. M. Vargas, M. A. Islas, J. Pacheco, G. J. Gutierrez, J. A. Meda-Campaña, D. Mujica-Vargas, and C. Aguilar-Ibañez, "Transformed structural properties method to determine the controllability and observability of robots," *Applied Sciences*, vol. 11, no. 7, p. 3082, 2021.

[14] D. I. Martinez, J. d. J. Rubio, A. Aguilar, J. Pacheco, G. J. Gutierrez, V. Garcia, T. M. Vargas, G. Ochoa, D. R. Cruz, and C. F. Juarez, "Stabilization of two electricity generators," *Complexity*, vol. 2020, 2020.

[15] M. Heryanto, H. Suprijono, B. Y. Suprapto, 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.

[16] G.-Y. Yoon, A. Yamamoto, and H.-O. Lim, "Mechanism and neural network based on pid control of quadcopter," in *2016 16th International Conference on Control, Automation and Systems (ICCAS)*, pp. 19–24, IEEE, 2016.

[17] Z. Tahir, M. Jamil, S. A. Liaqat, L. Mubarak, W. Tahir, and S. O. Gilani, "Design and development of optimal control system for quad copter uav," *Indian Journal of Science and Technology*, vol. 9, no. 25, pp. 10–17485, 2016.

[18] H. Housny, H. El Fadil, *et al.*, "Pso-based anfis for quadrotor system trajectory-tracking control," in *2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, pp. 1–6, IEEE, 2020.

[19] B. Selma, S. Chouraqui, and H. Abouaïssa, "Hybrid anfis-ant colony based optimisation for quadrotor trajectory tracking control," *International Journal of Modelling, Identification and Control*, vol. 34, no. 1, pp. 13–25, 2020.

[20] M. Al-Fetyani, M. Hayajneh, and A. Alsharkawi, "Design of an executable anfis-based control system to improve the attitude and altitude performances of a quadcopter drone," *International Journal of Automation and Computing*, pp. 1–17, 2020.

[21] M. B. Hossen and B. C. Ghosh, "Performance analysis of a pmbldc motor drive based on anfis controller and pi controller," in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pp. 1–6, IEEE, 2019.

[22] S. Khatoon, I. Nasiruddin, and M. Shahid, "Design and simulation of a hybrid pd-anfis controller for attitude tracking control of a quadrotor uav," *Arabian Journal for Science and Engineering*, vol. 42, no. 12, pp. 5211–5229, 2017.

[23] B. Selma, S. Chouraqui, and H. Abouaïssa, "Optimal trajectory tracking control of unmanned aerial vehicle using anfis-ipso system," *International Journal of Information Technology*, vol. 12, no. 2, pp. 383–395, 2020.

[24] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics and Automation magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[25] T. L. Blevins, "Pid advances in industrial control," *IFAC Proceedings Volumes*, vol. 45, no. 3, pp. 23–28, 2012.

[26] N. Aphiratsakun and N. Otaryan, "Ball on the plate model based on pid tuning methods," in *2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp. 1–4, IEEE, 2016.

[27] P. Meshram and R. G. Kanojiya, "Tuning of pid controller using ziegler-nichols method for speed control of dc motor," in *IEEE-international conference on advances in engineering, science and management (ICAESM-2012)*, pp. 117–122, IEEE, 2012.

[28] A. A. Azman, M. H. F. Rahiman, N. N. Mohammad, M. H. Marzaki, M. N. Taib, and M. F. Ali, "Modeling and comparative study of pid ziegler nichols (zn) and cohen-coon (cc) tuning method for multi-tube aluminum sulphate water filter (mtas)," in *2017 IEEE 2nd International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, pp. 25–30, IEEE, 2017.

[29] S. Anusha, G. Karpagam, and E. Bhuvaneswarri, "Comparison of tuning methods of pid controller," *International Journal of Management, Information Technology and Engineering*, vol. 2, no. 8, pp. 1–8, 2014.

[30] T. Aleksei, P. Eduard, and B. Juri, "A flexible matlab tool for optimal fractional-order pid controller design subject to specifications," in *Proceedings of the 31st Chinese Control Conference*, pp. 4698–4703, IEEE, 2012.

[31] L. A. Zadeh, "Fuzzy sets as a basis for a theory of possibility," *Fuzzy sets and systems*, vol. 1, no. 1, pp. 3–28, 1978.

[32] J.-S. Jang, "Anfis: adaptive-network-based fuzzy inference system," *IEEE transactions on systems, man, and cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.

[33] M. Şahin and R. Erol, "A comparative study of neural networks and anfis for forecasting attendance rate of soccer games," *Mathematical and Computational Applications*, vol. 22, no. 4, p. 43, 2017.

[34] J. S. Arora, *Introduction to optimum design*. Elsevier, 2004.

[35] H. K. Tran and T. N. Nguyen, "Flight motion controller design using genetic algorithm for a quadcopter," *Measurement and Control*, vol. 51, no. 3-4, pp. 59–64, 2018.

[36] S. Mirjalili, "Genetic algorithm," in *Evolutionary algorithms and neural networks*, pp. 43–55, Springer, 2019.

[37] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation*, vol. 5, pp. 4104–4108, IEEE, 1997.

[38] R. Hassan, B. Cohanim, O. De Weck, and G. Venter, "A comparison of particle swarm optimization and the genetic algorithm," in *46th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference*, p. 1897, 2005.

[39] C. Ou and W. Lin, "Comparison between pso and ga for parameters optimization of pid controller," in *2006 International conference on mechatronics and automation*, pp. 2471–2475, IEEE, 2006.