Attitude control of Quadrotor using enhanced intelligent Fuzzy PID controller

Michael Azer[1], Dr. Abdulla Ismail[2]

[1]Graduate Student, Dept. of Electrical Engineering, Rochester Institute of Technology, Dubai, UAE
[2]Professor, Dept. of Electrical Engineering, Rochester Institute of Technology, Dubai, UAE

Abstract - *Over the past few years, the increase of demand on drones have started to affect human life, be it civilian or military. This vast involvement of drones/quadrotors demanded a better flight control performance to ensure accuracy and reliability of the task given to the drone. The main goal of this study is to show the improvement in stabilizing the flight attitude of the quadrotor system. At first, PID controllers were perceived as the most reliable type of controllers and therefore used in this field as it achieves the system stability within an acceptable performance. However, the main challenge for the PID controller was to find the right gains so that it can reach stability faster with a less settling time and near zero steady state error. Therefore, in this paper the Fuzzy PID controller is proposed as a replacement for the traditional PID. Since fuzzy control is able to provide better adaptive gains to achieve improved stability performance of the quadrotor. To prove the legitimacy of the Fuzzy controller over the PID, a comparison study was carried to demonstrate the performance methods of the fuzzy controlled quadrotor and conclude the improvement over the traditional PID.*

Key Words: UAV, Quadrotor, PID, Fuzzy PID, Drones, Matlab, PID tuning.

1. INTRODUCTION

We get to hear the word "drone" in many fields and a lot of times in the news, which proves its involvement and raised interest in many different fields. It is introduced as a small aircraft with no pilot —from little kids remotely control toys, to autonomous flying machine and advanced complicated military weapon models. The reason is that the word Drone is defined differently in each situation.

A drone is defined as: "an unmanned aircraft or ship guided by remote control or on board computers". This definition gives a very large sense of the word, which provides to the over-generalizations and misinformation we see when the media reports on a particular type of unmanned aircraft.[1].

1.1 Important definitions

UAV

Unmanned Aerial Vehicles (UAV) are autonomous flying drone, which is basically being programmed to do a specific task without a pilot or constantly being monitored. This autonomous element adds many advantages to the usage of UAVs such as reduce human error, reduce expenditure and increase its ability to maneuver through sharp trajectories.

Roll

This movement is done by rotating motor 1 with faster (slower) speed and rotating motor 3 with slower (faster) speed as shown in the Figure 1.

Pitch

This movement is done by rotating motor 2 with faster (slower) speed and rotating motor 4 with slower (faster) speed, to rotate around Y axis as shown in the Figure 1.

Yaw

This movement around Z-axis, it is done by rotating motor 1 and 3 with faster (slower) speed or rotating motor 2 and 4 with faster (slower) speed, as shown in the Figure 1.

Forward/Backward

It results due to any changes occur in the pitch angle.

Right/Right Motion

It results due to any changes occur in the roll angle.

Take-Off/Landing

This movement is a result by rotating all motors with faster (slower) speed with the same average which increase (decrease) the vertical forces (thrust) which make the quadrotor raises (falls).

Hovering

This motion is a result by rotating the four motors with constant speed, which give thrust equal to the gravitational force so the quadrotor could fly at a fixed altitude. [1]

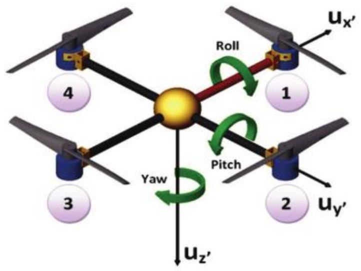


Fig -1: Pitch, Yaw and Roll of quadrotor

1.2 History of UAV

The UAV history has been through many improvements from the start in the ancient Greek period till now. It took man 2500 years to come up with the idea of flying until reaching to concocting an image of drones in our minds today. In the beginning, a lot of designs and ideas did not work until they found the efficient one. These advancements involved many aspects over a wide range of engineering fields such as electronics, communication, control and intelligent software. [1]

1.3 Advantages and Applications of UAS

The most obvious advantage of UAS is that it allows exploring and reaching new places without risking any casualties because these aircrafts do not need operators onboard.

Before looking into UAS in more detail, it is appropriate to list some of its important applications, which are as follows [2]:

Civilian application: Media recording, agriculture filed, helping coastguard, power providing companies, fire services and forestry, media news service, security Authorities

Military application: Navy, Army, Air Force

2. QUADROTOR MODELING

The quadrotor used in this paper is a rigid cross body which having four rotors in each corner of the frame, which are fixed-pitch-angle blades, different from the well known helicopters. However, some of the quadrotors have variable-pitch-angle blades of the rotors. The quadrotor here has four inputs and six outputs as shown in Figure 2

The quadrotor is controlled by changing the thrust forces and movements generated by each rotor.

2.1 Euler Angles

The inertial frame is transformed into the body frame using the rotations and translations. Forces and torques acting on the quadrotor are also evaluated in the body frame .The inertial frame, $F_{interial}$ = (X_I, Y_I, Z_I), is an earth-fixed coordinate system with its origin located conveniently at the base station. It is the conventional North, East and Down (NED) frame.

The body frame, $F_{bodyframe}$ = (X_b, Y_b, Z_b), has its origin at the Center of gravity of the Quad rotor with its x-axis pointing forward along motor 1, y-axis pointing out to right along motor 2 and z-axis pointing out the belly. The vehicle-carried NED frame, $F_{vehicle}$ = (x_v, y_v, z_v), has its axes aligned with the inertial NED frame but has its origin at the COG of the Quadrotor

2.2 Dynamics of quadrotor

A fixed frame body is assumed to be at the quadrotor's center of gravity if the z-axis is pointing upwards. This axis of the body is related to the inertial frame by a position vector (x, y, z) and 3 angles of Euler, pitch, roll and yaw, respectively.

The model structure is based on rotational and translational dynamics as shown in Figure 2. The rotational dynamics is used to calculate (ϕ,θ,ψ,) and translational dynamics to calculate (x,y,z) .[3]

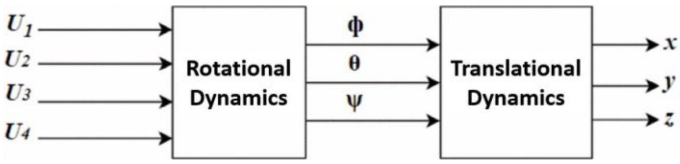


Fig -2: Input and Outputs of Dynamics of a quadrotor

2.3 Translational Dynamics

The linear equations of motion are explained in the global frame. To find the acceleration of the quadrotor we get the summation of the force of gravity, the thrust force of the motors, and the drag force. The thrust vector in the body frame is multiplied with the rotation matrix explained above to change it into the global frame using the rotation matrix described above. The total force is a result of the four motors in the global coordinate frame as defined below in equation (1).[4]

$$m\ddot{\mathbf{X}}^G = F_g - F_T^G - F_d \quad (1)$$

2.4 Rotational Dynamics

The equations of motion are governed by factors that limits the dynamic movement of the UAV, where the rotation of the quadrotor is calculated over the center of mass of the quadrotor. The inertia is perceived to be symmetrical from the assumption that the quadrotor is also symmetrical. The equations of motion are mainly dependent on the motor torques of the propellers and the aerodynamic effects. To maintain a stable flight, torque due to the drag is considered to be directly proportionate to the square of the angular velocity and the drag coefficient.

The equation of motion is determined by finding the two components: translational and rotational equations. The rotational equation in the frame determined about the quadrotor's center and not the center of the global frame. [4]

First equation (2), the gyroscopic effect is caused from the rigid body rotation. Second equation (3), the motors rotation creates rolling, pitching, and yawing torque.

$$J_b\dot{\omega} = \tau_m - \tau_g - (\omega \times J_b\omega) \tag{2}$$

$$\mathbf{J_b} = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix} \quad \dot{\omega} = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} \quad (\omega \times \mathbf{J_b}\omega) = \begin{bmatrix} \dot{\theta}\dot{\psi}(J_z - J_y) \\ \dot{\psi}\dot{\phi}(J_x - J_z) \\ \dot{\theta}\dot{\phi}(J_y - J_x) \end{bmatrix} \tag{3}$$

Where J_b is the quadrotor body moment of inertia

ω is the rate of angular velocity.

2.5 Equations of Motion

The equations of motion of quadrotor are representing below as two parts the translational dynamics in equation (4) and Rotational dynamics in equation (5).

$$\begin{bmatrix} \dot{X}^G \\ \dot{Y}^G \\ \dot{Z}^G \end{bmatrix} = \begin{bmatrix} cos(\psi)cos(\theta) & cos(\psi)sin(\phi)sin(\theta) - cos(\phi)sin(\psi) & cos(\phi)cos(\psi)sin(\theta) + sin(\phi)sin(\psi) \\ sin(\psi)cos(\theta) & sin(\phi)sin(\psi)sin(\theta) + cos(\phi)cos(\psi) & cos(\phi)sin(\psi)sin(\theta) - cos(\psi)sin(\phi) \\ -sin(\theta) & cos(\theta)sin(\phi) & cos(\phi)cos(\theta) \end{bmatrix} \begin{bmatrix} \dot{x}^b \\ \dot{y}^b \\ \dot{z}^b \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{1}{J_x}\left[(J_y - J_z)qr - J_r q(\omega_1 - \omega_2 + \omega_3 - \omega_4) + \ell K_T(\omega_4^2 - \omega_2^2)\right] \\ \frac{1}{J_y}\left[(J_z - J_x)pr + J_r p(\omega_1 - \omega_2 + \omega_3 - \omega_4) + \ell K_T(\omega_1^2 - \omega_3^2)\right] \\ \frac{1}{J_z}\left[(J_x - J_y)pq + K_d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2)\right] \end{bmatrix} \tag{5}$$

Below equations (6) are the final dynamical model equations of the quadrotor.

$$\ddot{x} = \frac{U_1(c\theta c\varphi) + U_1(s\theta c\phi c\varphi + s\phi s\varphi) - k_x\dot{x}}{m}$$
$$\ddot{y} = \frac{U_1(c\theta c\varphi) + U_1(s\theta c\phi s\varphi - s\phi c\varphi) - k_y\dot{y}}{m}$$
$$\ddot{z} = \frac{U_1(-s\theta) + U_1(c\phi c\theta) - k_z\dot{z} - g}{m} \tag{6}$$
$$\ddot{\phi} = \frac{(J_y - J_z)\dot{\theta}\dot{\psi} - J_r\dot{\theta}\Omega_{11} + LU_2}{J_x}$$
$$\ddot{\theta} = \frac{(J_z - J_x)\dot{\phi}\dot{\psi} - J_r\dot{\phi}\Omega_{11} + LU_3}{J_y}$$
$$\ddot{\varphi} = \frac{(J_x - J_y)\dot{\phi}\dot{\theta} + fU_4}{J_z}$$

As of the equations (7):

U (1) –is the four rotors thrust (altitude)

U (2) – is the motors are controlling the Roll which is the movement in x axis

U (3) – is the motors are controlling the Pitch which is the movement in y axis

U (4) – is the motors are controlling the Roll which is the movement in Z axis

The inputs are mapped as

$$U_1 = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2)$$
$$U_2 = b(-\Omega_2^2 + \Omega_4^2) \tag{7}$$
$$U_3 = b(\Omega_1^2 - \Omega_3^2)$$
$$U_4 = d(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2)$$

Where Ω_i (i=1,2,3,4) is the angular speed of the motor.

2.6 Quadrotor Model Characteristics

The characteristics and assumptions that made the model easier for calculations:

•The body frame is assumed to be rigid and Symmetrical

•Assume the center of the body frame always coincides with the center of gravity of the body.

•Thrust and drag are directly proportional to the square of rotor's speed.

2.7 Quadrotor modeling using MATLAB

As we discussed in the above section on the quadrotor modelling equations, the Simulink model below in Figure 3 shows the quadrotor model that consists of the following main block diagram:

•rotational dynamics used to compute angular accelerations

•angular velocities are feeding the translational dynamics with the angles

•translational dynamics used to calculate the translational acceleration

•motor speed calculator is limiting the control inputs to the physical motor parameter

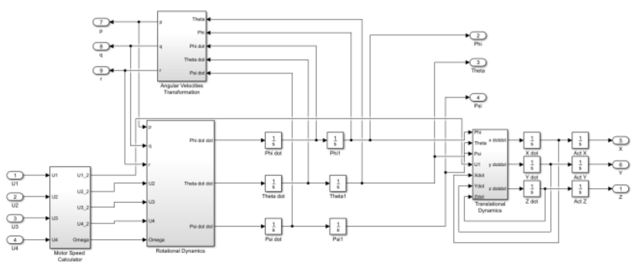


Fig -3: Simulink model of quadrotor system

3. PID CONTROL OF QUADROTOR

The quadrotor are made of four motors and the synchronization between the motors by speed and the direction that make the movement. The problem is to find a suitable controller for each electric motor of the quadrotor.

3.1 Introduction to PID Controller

PID stands for Proportional- Integral-Derivative control. The three functions of the PID are used to change the system performance factors such as the steady state and the transient responses specific conditions. Proportional-integral-derivative controller is considered as one of the most efficient and effective methods in control problems as it offers a simple, economic and reliable solution to control engineering challenges.

PID was used for the first time, in 1910 by Elmer Sperry to control ship steering and later added many methods of tuning to get better performance like the heuristic method in 1942 called Ziegler–Nichols' (Z-N). Nowadays, the PID algorithms is taking the biggest part of industrial controllers, especially at low level controls, because the main properties of the PID algorithms are simple, easy to use , applicable and good performance. Its wide applications has fortified and supported the improvement of different PID tuning strategies, modern computer program bundles, and equipment modules. [5]

The standard PID transfer function is commonly known in the "parallel form" or the "ideal form" as in equation (8):

$$G(s) = K_P + K_I \frac{1}{s} + K_D s = K_P \left(1 + \frac{1}{T_I s} + T_D s\right) \quad (8)$$

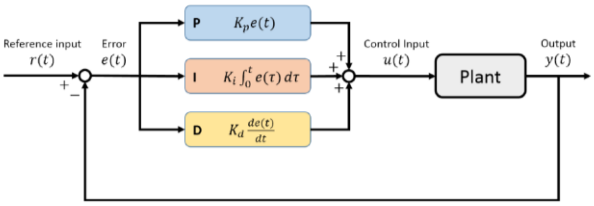


Fig -4: PID controller

Each of the three functions of the PID controller affects the performance of the system. Thus, the tuning of the three parameters is crucial to achieve the optimal performance. Table 1 shows the effects of the three functions individually on the system performance.

Table -1: The PID parameters effect

Closed-Loop Response	Rise Time	Overshoot	Settling Time	Steady-State Error	Stability
Increasing K_P	Decrease	Increase	Small Increase	Decrease	Degrade
Increasing K_I	Small Decrease	Increase	Increase	Large Decrease	Degrade
Increasing K_D	Small Decrease	Decrease	Decrease	Minor Change	Improve

3.2 Quadrotor Model after using PID

The mathematical equation of a PID controlled quadrotor can be represented as a set of well-integrated equations to work simultaneously and successfully to control the flight parameters of a quadrotor.

The reason to implement the position controller is to prevent the translational drifting behavior. The position controller inputs are the position measurements and the outputs are the desired roll and pitch angles to move the quadrotor to the desired x and y position. The Altitude controller is implemented to stabilize the quadrotor hover.

The PID controller is used by the quadrotor model to control the angular, altitude and translational positions as shown in Figure 5

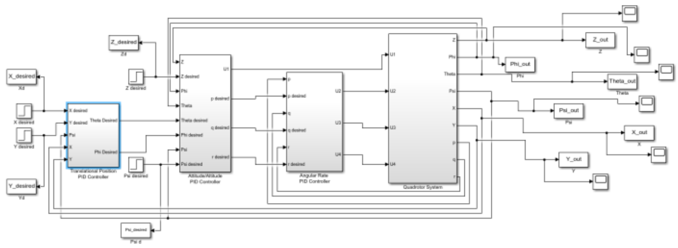


Fig -5: Simulink model of quadrotor system

PID control of roll motion:

The equation to calculate the input of controller for the roll (U2) is given below in (8). This is demonstrated in the Simulink model as shown in Figure 6.

$$u_2(t) = k_P e_p(t) + k_I \int_0^t e_p(t)dt + k_D \frac{e_p(t) - e_p(t-1)}{dt} \quad (8)$$

where ep(t) is error difference between desired p and actual p roll.

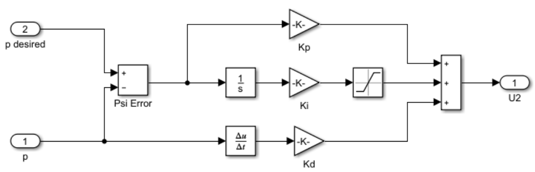


Fig -6: Roll PID controller

PID control of pitch motion:

The equation to calculate the input of controller for the pitch (U3) is given below in equation (9). This is demonstrated in the Simulink model as shown in Figure 7.

$$u_3(t) = k_P e_q(t) + k_I \int_0^t e_q(t)dt + k_D \frac{e_q(t) - e_q(t-1)}{dt} \quad (9)$$

where $e_q(t)$ is difference between desired q and actual q pitch.

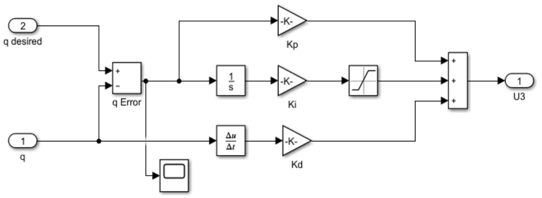


Fig -7: Pitch PID controller

PID control of yaw motion:

The equation to calculate the input of controller for the yaw (U4) is given below in equation (10). This is demonstrated in the Simulink model as shown in Figure 8.

$$u_4(t) = k_P e_r(t) + k_I \int_0^t e_r(t)dt + k_D \frac{e_r(t) - e_r(t-1)}{dt} \quad (10)$$

where er(t) is difference between desired r and actual r yaw.

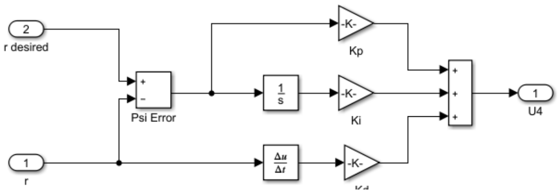
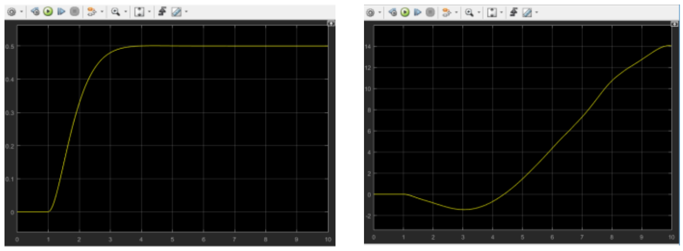


Fig -8: Yaw PID controller

3.4 Quadrotor PID controller Simulation

After running the Simulink, the graphs below is showing the attitude, Pitch, Roll and YAW of the quadrotor as it reached steady state.

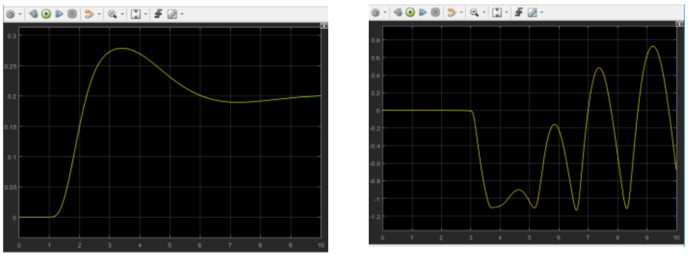
As we can see in Figure 9, after adding 0.5 step input , the settling time of the altitude became 3.8 seconds and there is no steady state error for the PID controller. The open loop system required almost 10 seconds for the system to settle down.



Altitude response with PID controller | Altitude response for open loop system

Fig -9: Altitude response (amplitude vs time)

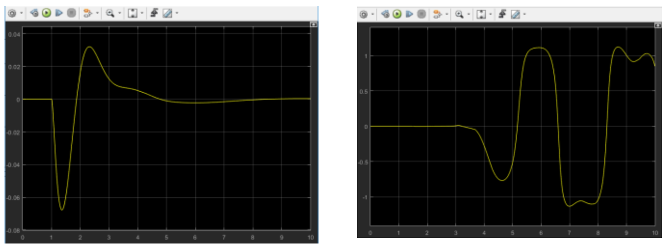
Observing the roll response in Figure 10, we found the settling time to be around 4 seconds without any steady state error. However, in the open loop the system is unstable.



Roll response with PID controller | Roll response for open loop system

Fig -10: Roll response (amplitude vs time)

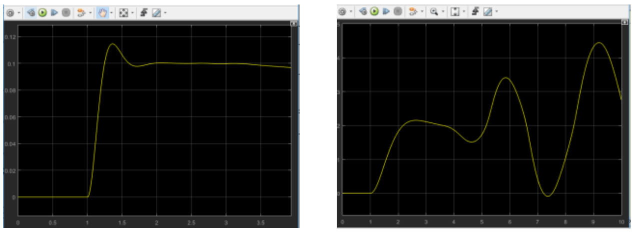
Observing the pitch response in Figure 11, we found the settling time to be around 9 seconds with no steady state error. However, the open loop system is unstable.



Pitch response with PID controller | Pitch response for open loop system

Fig -11: Pitch response Amplitude vs time

Observing the Yaw response in Figure 12, we found the settling time around 2 seconds with no steady state error. However, in the open loop system is unstable.



Yaw response with PID controller   Yaw response for open loop system

Fig -12: Yaw response (Amplitude vs time)

As it is shown that, the quadrotor system with PID controller can reach the stability after a specific time period, although, the open loop system shows instability response. The PID controller is known to be a good controller as it is used to get the desired results. But it could still be improved if the gain is well tuned and that will be discussed in the following topic.

4. Fuzzy Control of Quadrotor

Fuzzy control (FL) is a global long-term practical experience obtained as a result introducing adaptive intelligent algorithms, making the control effect ideal trying to mimics human thinking logic to control certain plants/operations. As FL has an added adaptive distinct feature which allows it to achieve better control outputs to any given input, making it able to respond accordingly to any uncertainties to maintain the stability of the system

Fuzzy Logic Controller (FLC) is important for its ability to control nonlinear systems due its adaptable, robust and optimal implementations in a complex system.

4.1 Introduction to Fuzzy logic

4.1.1 History of Fuzzy logic

The brain of a human being is constantly taking decisions in all different situations. For example, while driving a car we are to assess the situation and deciding whether to use the brakes to stop the car or the accelerator to move it accordingly. Therefore, decision-making is a part of our day-to-day life and everyday situations which are hesitant and fuzzy.

Studies on unstable processes and muddy data entered a new age as published by Lotfi A. Zadeh in his article "Fuzzy sets". Although the paper was published in 1965, the use of fuzzy logic (FL) grew in the second half of the 1970s when Lotfi A. Zadeh published two more studies in which the application of fuzzy logic to uncertain systems and decision-making laid new theories. [6]

4.1.2 Applications of Fuzzy Logic

FL is used in many different fields, some of its most important applications in our real life are implemented in satellite technologies and space craft altitude control, whereas in the automotive field it is used to control speed , automated motion, and traffic control management.[6]

Fuzzy logic also has a very important role in NLP (Natural Language Processing), image processing, signal aliasing and in artificial intelligence including decision making and support systems for companies and businesses. Finally, it is taking a big share in the modern control system like flow control, motion control, temperature control and many other products.

4.1.3 Importance of Fuzzy Logic

Fuzzy logic (FL) controller is unlike the classic PID (proportional integral derivative) controllers, it does not require mathematical model of the systems to control it. It changes the system input to achieve the desired output, by checking the current form the plant output. The fuzzy logic decision making process is made by controller depending on the design of the fuzzy sets and rules.

Due to its adaptive nature, the importance of FL has increased in the last few years, due to the variety of uses with linear or nonlinear systems.

Although the linearized system, with one of the approximation methods, reduces the system accuracy and shows more operation errors on the system, these errors still don't affect the performance of FL controllers as the mathematical model is not required and that is why the FL is one of the most important alternatives to controlling the nonlinear systems. [7]

4.1.4 Advantages of Fuzzy Logic

One of the advantages of Fuzzy Logic is its simple architecture that make it understandable and easy to modify as shown in Figure 13, which is made up of distinct processes being integrated together such as (1) Fuzzification, (2) Fuzzy inferencing and (3) Defuzzification.. The fuzzy logic algorithm can work with any type of input, even the noisy ones.

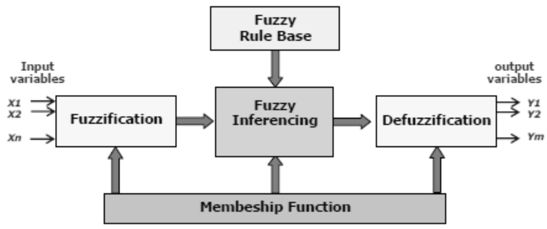


Fig. Elements of Fuzzy System

Fig -13: Fuzzy logic Process

4.2 Fuzzy logic process

The Fuzzy logic architecture consists of three main building blocks that require to be designed correctly to control a given system as in Figure 13.

4.2.1 Fuzzification

It is used to convert the crisp inputs (from sensors like temperature, pressure etc.) to fuzzy sets. There are many types of curves and tables used like the Gaussian and the trapezoidal membership function (MF). The MFs are graphical representations described mathematically with many parameters, so that the parameters and the shape of the MFs is adapted to reach the best performance of the FLC. Any particular input is interpreted from this fuzzy set, and a degree of membership is obtained as shown in Figure 14. Overlapping between the membership function is important to reach the smooth transition from a MF to another within the system operation.

Figure 14 is for an air-conditioning system input connected with temperature sensor. FL is applied to keep the room within an accepted temperature and to change the vent orientation accordingly. Therefore, the system changes between fuzzy sets cold, cool, good, warm and hot assessing its current/actual condition and deciding what it is the most optimal decision to be taken at every stage.

The Fuzzy logic of this example is embedded within the microprocessor that controls the speed of the fan motor. In order to understand how it works, we need to compare it with the human brain where a rational decision of changing the motor speed or vent directions depending on the current situation and the goal/required situation is executed.

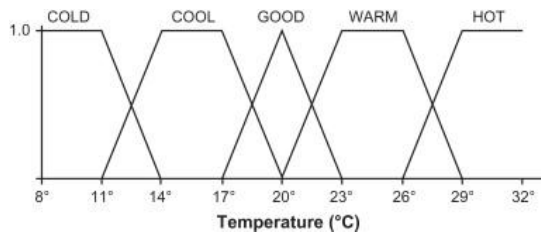


Fig -14: Example for Fuzzification

4.2.2 Fuzzy Rule Base

Fuzzy rule base is the part of the process which gives the condition to reach the decision making. It gives it an "if –then" condition. Many improvements have been brought to the fuzzy rules to make it shorter and easier. The importance of the fuzzy rule base is that the output accuracy depends on it as it gives rule conditions to each variable in the system. It defines the input with a set of conditions, the output with the other set of output conditions and combines the input with output together. It is the brain that gives rules to the controller based on which the most rational/optimal decisions are carried out. Another important factor in the calculation of controller efficiency is the number of Fuzzy rules.

4.2.3 Defuzzification

Defuzzification is the reverse process of fuzzification. It is used to convert fuzzy sets into a crisp output value. The main thing is to change the crisp numeric value that results from fuzzy value of the linguistic variable. There are many methods of defuzzification. The one used here is the centroid method, as the name is derived from the center of gravity.

4.2.4 A Mamdani Fuzzy Logic method

It is one of the most common and well know fuzzy methods. It is good for manual inputs, it accepts wide spread input. It is also easy to use and understand. Mamdani fuzzy systems use fuzzy sets as rule consequent. Therefore, this method will be used in this paper. [6]

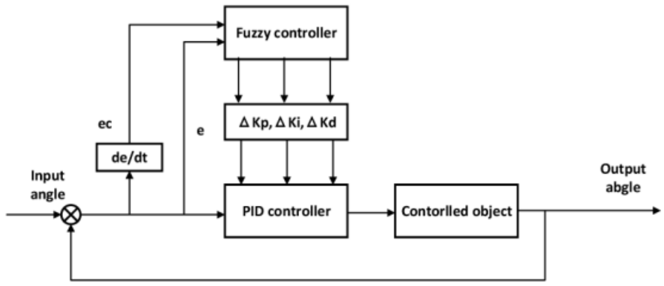


Fig -15: Fuzzy control Design of for Quadrotor system

In the fuzzy control of Quadrotor model, the error (e) and change in error (ē) are used as inputs and PID gains (Δk_p, Δk_i, and Δk_d) are the system outputs see Figure 15. The main function of the fuzzy PID controller is to tune the PID by reaching to a relationship between input and output (parameters k_p, k_i, k_d and e, ē) that bring the system to stability within a set of constraints imposed by the dynamics of the quadrotor

The three PID parameters are adjusted in the actual operation of the system in compliance with the fuzzy control theory, to fulfill the control parameter requirements by continuous detection for e and ē. Therefore, we can use it as: [8]

$$\overline{k_p} = k_p + \Delta k_p$$

$$\overline{k_i} = k_i + \Delta k_i$$

$$\overline{k_d} = k_d + \Delta k_d$$

where k_p, k_i, k_d are preliminary parameters of PID controller. Δk_p, Δk_i, Δk_d are adjustable gains from the fuzzy controller. The fuzzy controller's foundation is Fuzzy Rules. Fuzzy rule design correctness can directly affect controller performance. Fuzzy rules can be demonstrated between e, ē and Δk_p, Δk_i, Δk_d.

4.4 Matlab modeling of Fuzzy PID controlled Quadrotor

The Simulink quadrotor model is connected to Fuzzy PID block as shown in Figure 16. The Fuzzy PID controller is used to control the pitch, roll, yaw and altitude to calculate the results of U1, U2, U3 and U4.

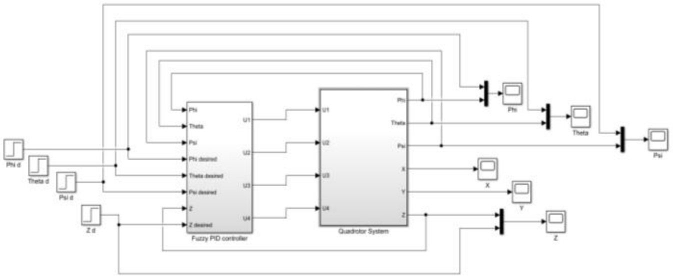


Fig -16: Roll Control with Fuzzy PID controller

In Figure 17, the Fuzzy PID controller is added with 2 inputs (difference in error) and 2 outputs (kp, kd) using the same equation for PID with roll equation.

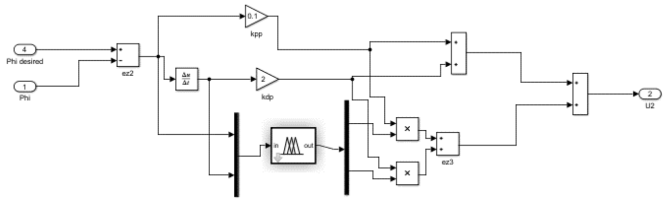


Fig -17: Roll control with Fuzzy PID controller

4.4.2 Pitch Control with Fuzzy PID controller

In Figure 18, Fuzzy PID controller is added to the system with 2 inputs and 2 outputs using the same PID equation with pitch equation.

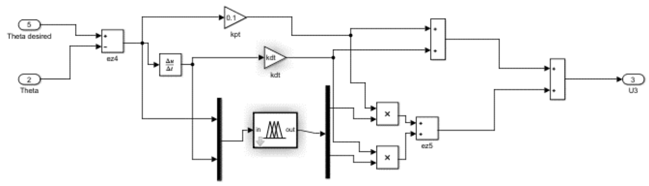


Fig -18: Pitch control with Fuzzy PID controller

4.4.3 Yaw Control with Fuzzy PID controller

In Figure 19, Fuzzy PID controller is added to the system with 2 inputs and 2 outputs using the same PID equation with yaw equation.

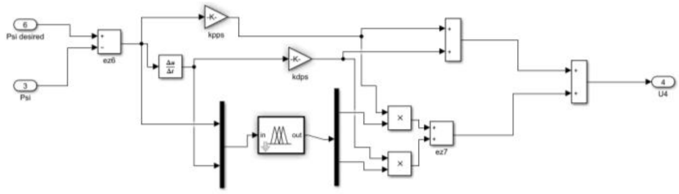


Fig -19: Yaw control with Fuzzy PID controller

The design of the fuzzy controller with the Matlab tool is shown in Figure 20. After many trials, we got an acceptable solution using this design. Firstly, we chose the method to use in controller and as we mentioned before, we will use Mamdami method for fuzzification. Then start adding the variables, in this case, 2 inputs and 2 outputs.

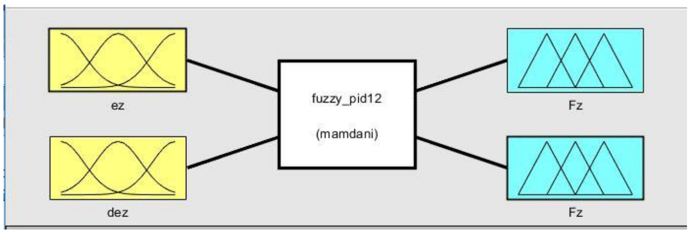


Fig -20: Design 2 inputs and 2 outputs for Fuzzy PID controller of quadrotor

Each variable added is needed to design the membership function and the range of the fuzzy sets. For the inputs we use the range from -2 to 2 and for the outputs we use the range from -15 to 15. The inputs membership function is chosen to be Gaussian as shown in Figure 21 and for the outputs the trapezoidal function is used, as shown in Figure 22.

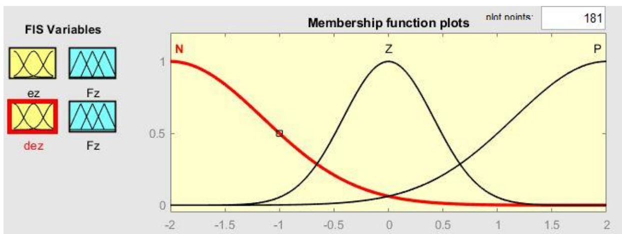


Fig -21: The input membership function

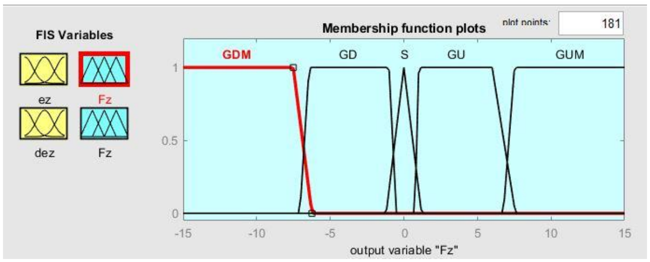


Fig -22: The output membership function

For the fuzzy rules, 15 rules are used as shown in Figure 23. Each rule base line shows the relation between the two inputs variables (error and the change of error) and the two outputs variables of the PID gains.

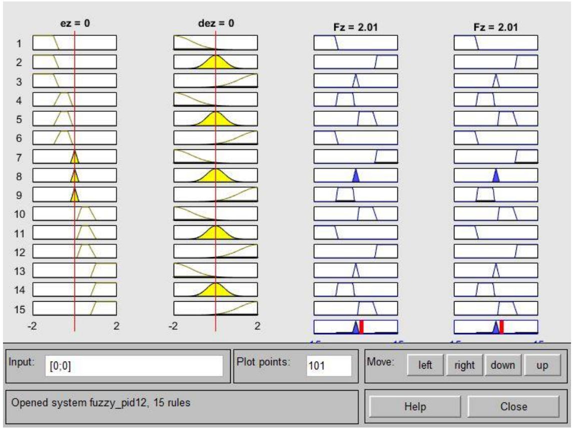


Fig -23: Fuzzy rules for the Fuzzy PID controller

4.5 Comparison of PID and Fuzzy PID controlled quadrotor

In the graphs below, simulation using Matlab code shows the response(pitch, roll, yaw and altitude)of the quadrotor using PID and Fuzzy PID controller by applying in 0.2 step input. The results are as follows:

For the roll response in Figure 24, we observe that the fuzzy tuned PID settling time is 4 seconds with no steady state error and for the PID controller the settling time is 10 seconds with no steady state error. Thus, we can conclude the Fuzzy PID controller provides a better performance for the roll control than PID controller.

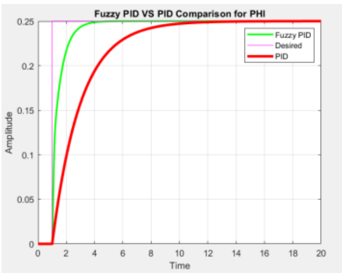


Fig -24: System Roll response (Fuzzy PID vs PID)

By looking at Figure 25, the pitch response shows that the fuzzy PID results is more efficient as the settling time is 4 seconds, although the PID needs more than 10 seconds to reach steady state.

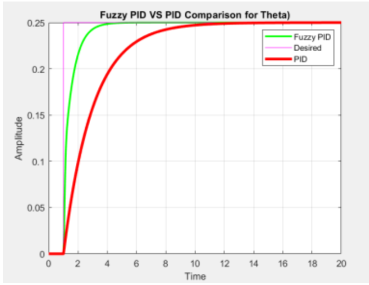


Fig -25: System Pitch response (Fuzzy PID vs PID)

By observing Figure 26, the yaw response shows that although the PID started to slightly get better, but it reached steady state in more than 20 seconds, whereas the Fuzzy PID controller settling was 11 seconds. Thus, we can conclude that the Fuzzy PID is giving better steady state response than the PID controller.

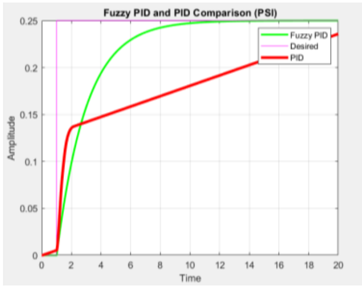


Fig -26: System Yaw Response (Fuzzy PID vs PID)

By observing the Z (altitude) response in Figure 27 we can see that settling time of the Fuzzy PID is 10 sec and the steady state error is zero. However the PID reaches the steady state and has a rather large steady state error.

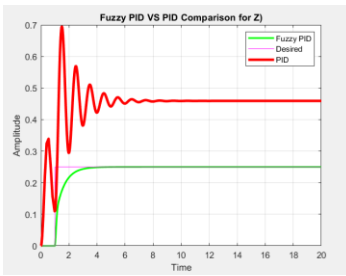


Fig -27: System Altitude Response (Fuzzy PID vs PID)

As shown in Table 2, the Fuzzy PID controller provided less settling time and smaller steady state error.

Table -2: Altitude PID and fuzzy PID responses specs

Time domain	PID Controller	Fuzzy PID Controller
Overshoot	0.45 secs	0.25 secs
Settling Time	9 secs	3.8 secs

Overall, the better performance shown by the Fuzzy PID is due to the better tuning of the controller gains that resulted better system performance. The fuzzy control acts like the brain of the PID to constantly tune and choose the values of the gains that bears the best results.

5. Conclusion, Contribution and future work

This paper studied the attitude of the quadrotor and how fuzzy logic was incorporated in the control of the overall attitude flight performance. After mathematically defining the system, the study of the PID controller and its enhancement of the quadrotor performance was observed and compared with an open loop or on controller system. The PID controller demonstrated an acceptable ability to control the quadrotor and bring it to stability with minimum steady state error. Separate controllers where used for each flight parameter (roll, pitch and yaw), so it would be easier to tune and thus perform better. These results were visible through the simulations that demonstrated the different effects of each controller on the system.

Secondly, a fuzzy tuned PID controller was introduced to control the quadrotor through the same flight condition of that were used in the PID simulations. The fuzzy logic controller was tailored to the quadrotor dynamics using fuzzification, rule base and defuzzification processes. It was obvious that the fuzzy PID performed better in all aspects especially the settling time and the steady state error.

With the involvement of UAV advancements in civilian, industrial and military fields, fuzzy logic has offered a solution to overcome the limitation of the traditional PID controller easily, giving it the ability to be more controllable.

Matlab simulation demonstrated how the Fuzzy tuned PID is more efficient than the classical PID as it is easier to reach the most optimal gains needed.

The contribution to this project was dedicated to mathematical model integration of the fuzzy logic controller, as well as implementing adaptive fuzzy logic algorithms in the quadrotor control. Also analyzing and assessing the output results of the different controllers which required critical understanding of the quadrotor dynamic system.

Future works could include the development of the following aspects that would guarantee better results for quadrotor control.

- Development of adaptive fuzzy through easier algorithms
- Different Quadrotor model design with better dynamics should be developed to improve the flight control
- Introducing new designs to UAV

REFERENCES

[1] Kilby, Terry, and Belinda Kilby. Make - Getting Started with Drones: Build and Customize Your Own Quadcopter, Maker Media, Inc, 2015.

[2] Dalamagkidis K. Aviation History and Unmanned Flight. In: Valavanis K., Vachtsevanos G. (eds) Handbook of Unmanned Aerial Vehicles. Springer, Dordrecht,2015.R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[3] Samir BOUABDALLAH, "Design and Control of quadrotors with application to autonomous flying Lausanne, 2007.

[4] Wil Selby. Quadrotor model. "A Record of My Tech Research and Travel Adventures." 2016.

[5] Ang, K.H. and Chong, G.C.Y. and Li, Y. PID control system analysis, design, and technology. IEEE Transactions on Control Systems Technology 13, 2015.

[6] Altas, Ismail H. Fuzzy Logic Control in Energy Systems: with Design Applications in MATLAB /Simulink. Institution of Engineering and Technology, 2017.

[7] Altaş, Ismail H. Fuzzy Logic Control in Energy Systems - With Design Applications in MATLAB®/Simulink® - 1.1 Introduction. Institution of Engineering and Technology, 2017.

[8] Huang, Tianpeng, et al. "Attitude Tracking for a Quadrotor UAV Based on Fuzzy PID Controller." 2018 5th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS), 2018.

BIOGRAPHIES

Michael Azer is a graduate of Electrical Engineering. Currently pursuing his Master of Science degree in Electrical Engineering, specializing in Control Systems, at Rochester Institute of Technology (RIT).

DrAbdulla Ismail obtainedhisB.Sc. ('80), M.Sc. ('83), and Ph.D. ('86) degrees, all in electrical engineering, from the University of Arizona, U.S.A. Currently, a professor of Electrical Engineering and assistant to the President at the Rochester Institute of Technology, Dubai, UAE.