Design of a Fuzzy-PID Controller for Controlling Quadrotor Attitude and Altitude

Mrinmoy Sarkar, Brian Baity, Dr. Homaifar

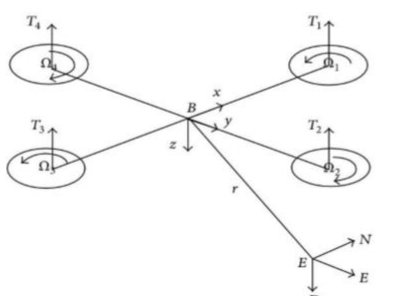
***Abstract*—The problem that we are faced with is designing a fuzzy controller to carry-out circular motion and hovering . Our foundation was built upon modeling and tuning a conventional PID controller to later on fuzzify this system. Once the system became fuzzy we added noise to compare the response of the conventional to the fuzzy controller.**

***Keywords:* Quadrotor, PID, fuzzy-PID, Simulink**

I. INTRODUCTION

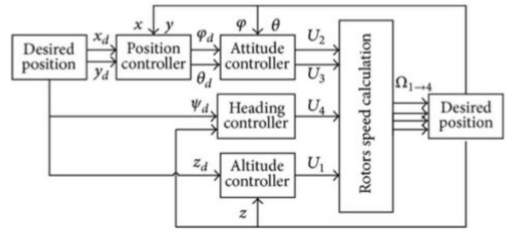
This assignment is geared towards furthering our knowledge of fuzzy systems and they can be applied to real world problems. Our problem is based upon the lateral movements, circular trajectory, and hovering of a quadrotor. The advantages of this fuzzy system is that it solves the problem of fixed gains, is cheaper to develop, covers a wide range of operating conditions are more readily customizable, and are robust in terms of uncertainty. The challenges we faced with this system is that is nonlinear, under actuated (lower number of actuators than degrees of freedom), low on board processing capability, low operation time and low efficiency in power consumption.

II. THEORETICAL REPRESENTATION



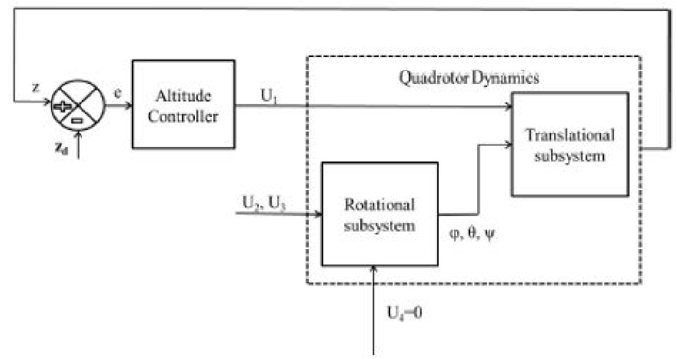
Structure of a quadrotor

Above is the model of a quadrotor it has four rotors that have the same size and radius and are symmetrically placed across two arms. The rotors that are positioned on the same arm rotate in the same direction i.e. rotor 2 and 4 rotate clockwise. B is the body coordinate system while E is the ground coordinate system.



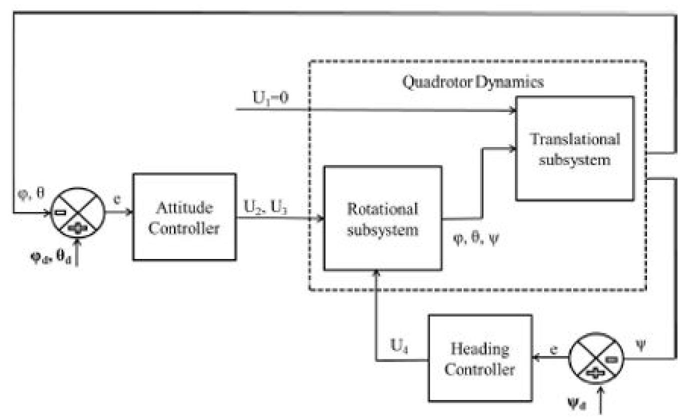
Quadrotor control structure

The Quadrotor control structure shows that the outer loop is for navigation and inner loop is designed to asymptotically track the desired attitude and altitude.



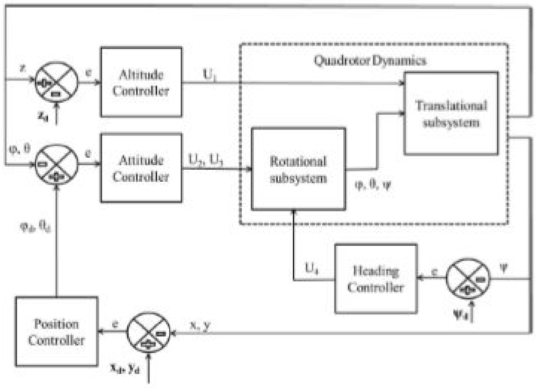
Block Diagram for Altitude Controller

The altitude controller uses the difference between the desired altitude and the actual altitude as an error signal for the input which produces a control signal U1.



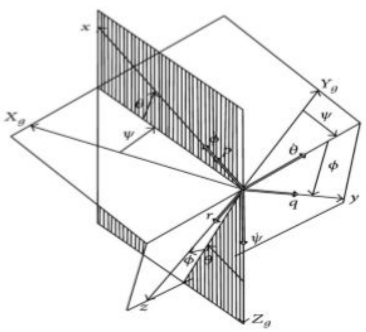
Block Diagram for Attitude and Heading Controller

The attitude and heading controller is similar to that of the previous altitude controller. It uses the difference between roll, yaw, and pitch and their actual values phi, theta and psi. This controller produces the output signals U2, U3, and U4.



Block Diagram for Position Controller

The position controller is different from the previous controllers. The x and y positions are not decoupled because they can't be controlled directly using one of the four control laws only. They are controlled through the roll and pitch angles. These desired angles are calculated from the translation equations of motion.



The relationship between the angular velocity components and the attitude angle change rate.

III. Mathematical Equations

Modeling of Quadrotor

$$R(x,\phi)=\begin{bmatrix}1 & 0 & 0\\ 0 & \cos\phi & \sin\phi\\ 0 & -\sin\phi & \cos\phi\end{bmatrix}$$

$$R(y,\theta)=\begin{bmatrix}\cos\theta & 0 & -\sin\theta\\ 0 & 1 & 0\\ \sin\theta & 0 & \cos\theta\end{bmatrix}$$

$$R(z,\psi)=\begin{bmatrix}\cos\psi & \sin\psi & 0\\ -\sin\psi & \cos\psi & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$R_{E\to B}=R(x,\phi)\,R(y,\theta)\,R(z,\psi)$$

$$R_{B\to E}=R^T_{E\to B}.$$

The rotation matrix from the ground to body coordinate system portrays rotation around the z-axis then rotation around the y-axis lastly rotation around the x-axis. This is where c. = cos(.) and s. = sin(.). Respectively roll angle, pitch angle and yaw angle around the x, y, and z axes.

$$\begin{bmatrix}p\\ q\\ r\end{bmatrix}=R_r\begin{bmatrix}\dot\phi\\ \dot\theta\\ \dot\psi\end{bmatrix},$$

where

$$R_r=\begin{bmatrix}1 & 0 & -\sin\theta\\ 0 & \cos\phi & \sin\phi\cos\theta\\ 0 & -\sin\phi & \cos\phi\cos\theta\end{bmatrix}$$

Angular velocity components p,q and r are the projection values on the body coordinate system of rotation angular velocity. The transformation matrix is based around a hovering position and Rr is assumed to be a unit matrix.

$$J\dot\omega+\omega\times J\omega+\omega\times[0\ \ 0\ \ J_r\Omega_r]=M_B$$

$$F_i=b\Omega_i^2$$

$$M_i=d\Omega_i^2$$

where Ω_i $(i=1,2,3,4)$ represents the ith rotor speed.

$$M_B=\begin{bmatrix}l\cdot b\left(-\Omega_2^2+\Omega_4^2\right)\\ l\cdot b\left(\Omega_1^2-\Omega_3^2\right)\\ d\left(\Omega_1^2-\Omega_2^2+\Omega_3^2-\Omega_4^2\right)\end{bmatrix}\qquad m\ddot r=[0\ \ 0\ \ mg]^T+RF_B\qquad F_B=\begin{bmatrix}0\\ 0\\ -b\left(\Omega_1^2+\Omega_2^2+\Omega_3^2+\Omega_4^2\right)\end{bmatrix}$$

The dynamics model is rotational and translational motions. While rotational movement is fully actuated, the translational movement is underactuated. In the body coordinate system, derivation of rotational movement equations are in respect to the law of momentum theorem and gyroscopic effect of the quadrotor.

$$\ddot\phi=\dot\theta\dot\psi\left(\frac{I_y-I_z}{I_x}\right)-\frac{J_r}{I_x}\dot\theta\Omega_r+\frac{L}{I_x}U_2,$$

$$\ddot\theta=\dot\phi\dot\psi\left(\frac{I_z-I_x}{I_y}\right)+\frac{J_r}{I_y}\dot\phi\Omega_r+\frac{L}{I_y}U_3,$$

$$\ddot\psi=\dot\phi\dot\theta\left(\frac{I_x-I_y}{I_z}\right)+\frac{1}{I_z}U_4,$$

$$\ddot x=-\frac{U_1}{m}\left(\cos\phi\sin\theta\cos\psi+\sin\phi\sin\psi\right),$$

$$\ddot y=-\frac{U_1}{m}\left(\cos\phi\sin\theta\sin\psi-\sin\phi\cos\psi\right),$$

$$\ddot z=g-\frac{U_1}{m}\left(\cos\phi\cos\theta\right),$$

The movement of the quadrotor rely on the above equations. The equations are based on the synthesizing of dynamics and kinematics models of the quadrotor. U1 through U4 are the input control variables.

$$
f(X,U) = \begin{bmatrix}
x_4 \\
x_5 \\
x_6 \\
-\dfrac{U_1}{m}\left(\cos x_7 \sin x_8 \cos x_9 + \sin x_7 \sin x_9\right) \\
-\dfrac{U_1}{m}\left(\cos x_7 \sin x_8 \sin x_9 - \sin x_7 \cos x_9\right) \\
g - \dfrac{U_1}{m}\left(\cos x_7 \cos x_8\right) \\
x_{10} \\
x_{11} \\
x_{12} \\
x_{11}x_{12}\dfrac{I_y - I_z}{I_x} - \dfrac{J_r}{I_x}x_{11}\Omega_r + \dfrac{L}{I_x}U_2 \\
x_{10}x_{12}\dfrac{I_z - I_x}{I_y} + \dfrac{J_r}{I_y}x_{10}\Omega_r + \dfrac{L}{I_y}U_3 \\
x_{10}x_{11}\dfrac{I_x - I_y}{I_z} + \dfrac{1}{I_z}U_4
\end{bmatrix}
$$

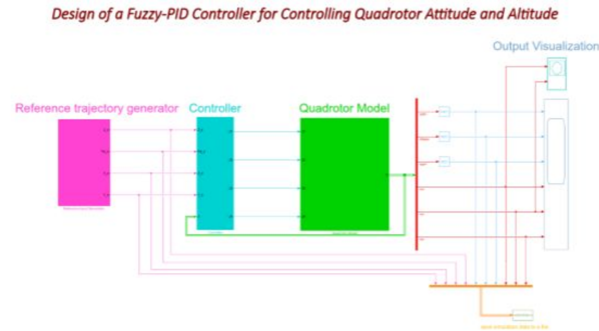
The above model is adopted by the control system is $\dot{X} = f(X, U)$. X is the state vector while U is the control input vector. The state vector is $X = [x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}\ \phi\ \theta\ \psi\ \dot{\phi}\ \dot{\theta}\ \dot{\psi}]^T$. In the controller design the state variables are $x1 = x$, $x2 = y$, $x3 = z$, $x4 = \dot{x}1 = \dot{x}$, $x5 = \dot{x}2 = \dot{y}$, $x6 = \dot{x}3 = \dot{z}$, $x7 = \phi$, $x8 = \theta$, $x9 = \psi$, $x10 = \dot{x}7 = \dot{\phi}$, $x11 = \dot{x}8 = \dot{\theta}$, and $x12 = \dot{x}9 = \dot{\psi}$. This is the result of synthesizing the motion equations of quadrotor, the state vector, and the control input variables.

$$GE = \frac{1}{max.error} \qquad GU = \frac{K_{df}}{GCE}$$

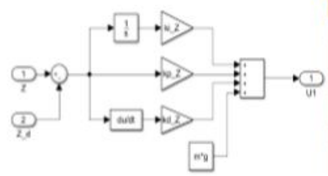
$$GCE = GE * \left(K_{pf} - \sqrt{K_{pf}^2 - 4K_{if}K_{df}}\frac{K_{if}}{2}\right) \qquad GCU = \frac{K_{if}}{GE}$$

The error normalization factor (GE) and the change in measurement normalization factor (GCE) measure the inputs received by the Fuzzy system. This factor is calculated by Kp−Fuzzy(Kpf), Ki−Fuzzy(Kif), Kd−Fuzzy(Kdf). The de-normalization factor (GU) and the change in the response of the de-normalization factor (GCU) measure the output responses of the Fuzzy system. This scaling factor is dependant upon gain information from the conventional PID.

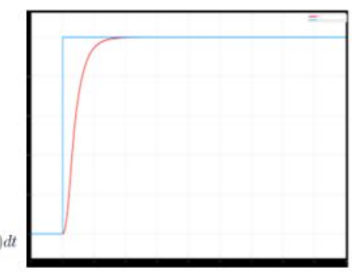
IV. Simulation



The Fuzzy-PID block diagram

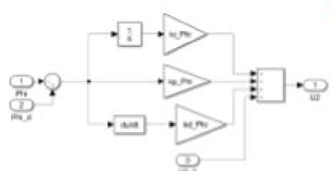


Parameter	Description	Value
k_p	Proportional gain	30
k_i	Integral gain	0
k_d	Derivative gain	10
t_r	Rising time	702.5 ms
t_s	Settling time	1.5 s
%OS	Overshoot	0

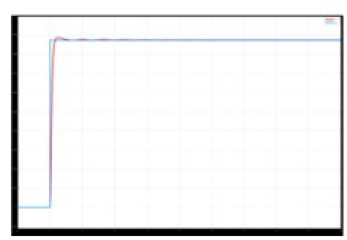


$$U_1 = k_p(z - z_d) + k_d(\dot{z} - \dot{z}_d) + k_i\int(z - z_d)dt$$

Altitude Controller

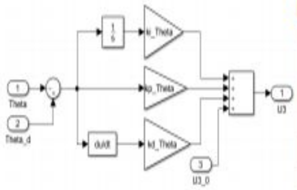


Parameter	Description	Value
k_p	Proportional gain	2.9996
k_i	Integral gain	0.17
k_d	Derivative gain	0.87
t_r	Rising time	110.1 ms
t_s	Settling time	597.5 ms
%OS	Overshoot	1.37



$$U_2 = k_p(\phi_d - \phi) + k_d(\dot{\phi}_d - \dot{\phi}) + k_i\int(\phi_d - \phi)dt$$

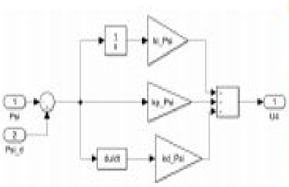
Roll Controller



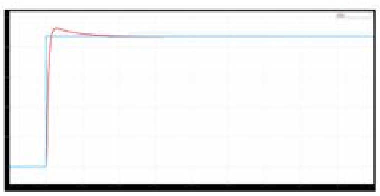
Parameter	Description	Value
k_p	Proportional gain	2.9996
k_i	Integral gain	0.17
k_d	Derivative gain	0.87
t_r	Rising time	110.1 ms
t_s	Settling time	597.5 ms
%OS	Overshoot	1.37

$$U_3 = k_p(\theta_d - \theta) + k_d(\dot{\theta}_d - \dot{\theta}) + k_i \int (\theta_d - \theta)dt$$

Pitch Controller

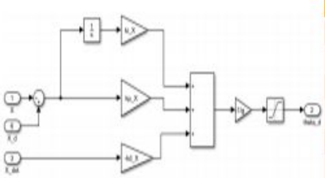


Parameter	Description	Value
k_p	Proportional gain	3.9
k_i	Integral gain	0
k_d	Derivative gain	2.57
t_r	Rising time	94 ms
t_s	Settling time	1.5s
%OS	Overshoot	6.09

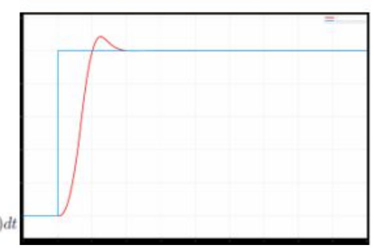


$$U_4 = k_p(\psi_d - \psi) + k_d(\dot{\psi}_d - \dot{\psi}) + k_i \int (\psi_d - \psi)dt$$

Yaw Controller

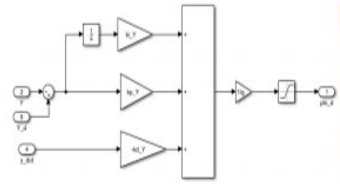


Parameter	Description	Value
k_p	Proportional gain	19.5
k_i	Integral gain	0
k_d	Derivative gain	5.7
t_r	Rising time	592.8 ms
t_s	Settling time	1.83 s
%OS	Overshoot	8.42

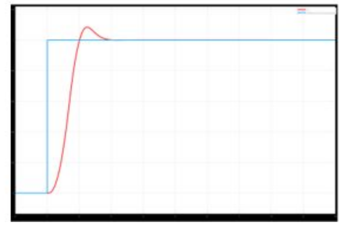


$$\ddot{x}_d = k_p(x_d - x) + k_d(\dot{x}_d - \dot{x}) + k_i \int (x_d - x)dt$$

Position Controller(X)

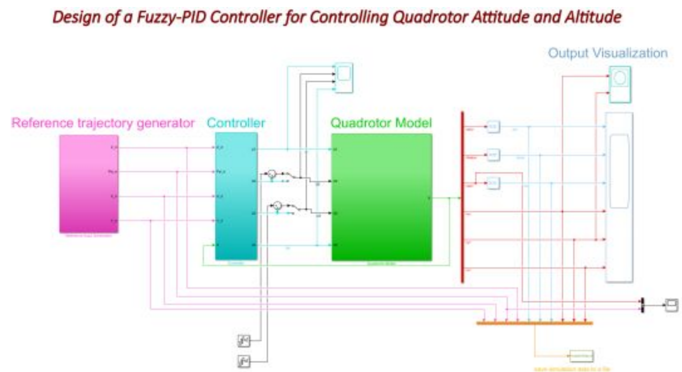


Parameter	Description	Value
k_p	Proportional gain	19.5
k_i	Integral gain	0
k_d	Derivative gain	5.7
t_r	Rising time	592.8 ms
t_s	Settling time	1.83 s
%OS	Overshoot	8.42

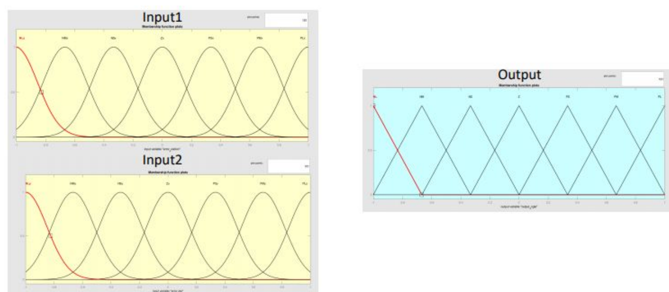


$$\ddot{y}_d = k_p(y_d - y) + k_d(\dot{y}_d - \dot{y}) + k_i \int (y_d - y)dt$$

Position Controller(Y)



Fuzzy-PID Controller generated (Added Noise)



Membership functions

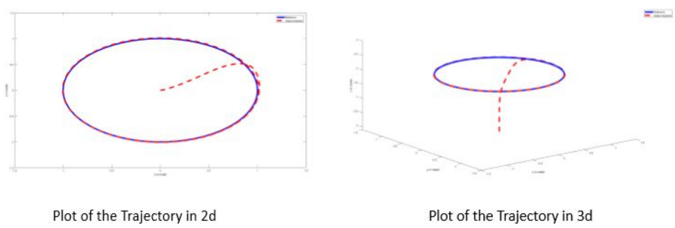
Two inputs and one output are normalized on the interval [-1, 1] with a scaling factor of 0.2 for every linguistic group. Input 1 takes the current angle obtained from the orientation sensor of the attitude. Input 2 is the error rate from the current angle t the reference angle.

In1/In2	NLy	NMy	NSy	Zy	PSy	PMy	PLy
NLx	NL	NL	NL	NL	NM	NS	Z
NMx	NL	NL	NL	NM	NS	Z	PS
NSx	NL	NL	NM	NS	Z	PS	PM
Zx	NL	NM	NS	Z	PS	PM	PL
PSx	NM	NS	Z	PS	PM	PL	PM
PMx	NS	Z	PS	PM	PL	PL	PL
PLx	Z	PS	PM	PL	PL	PL	PL

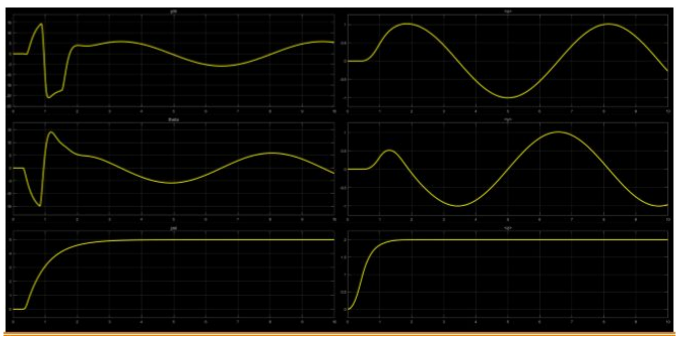
Linguistic Variable	Meaning
NL	Negative Large
NM	Negative Medium
NS	Negative Small
Z	Zero
PS	Positive Small
PM	Positive Medium
PL	Positive Large

Inference Rules and Linguistic variables(49 rules)

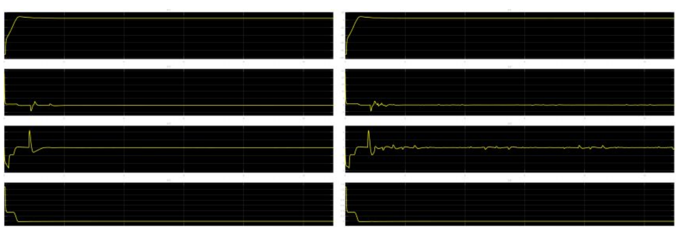
SIMULINK RESULTS



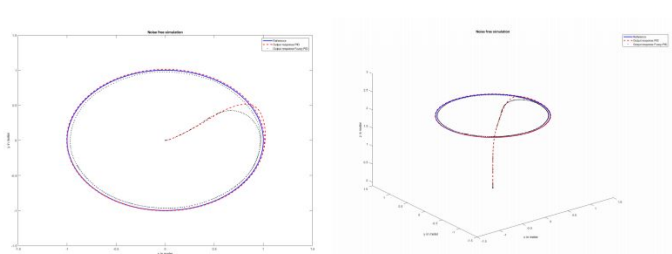
TRAJECTORY OF THE QUADROTOR IN 2D AND 3D(NO NOISE)



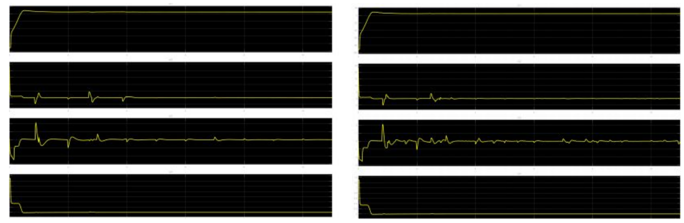
THE OUTPUT SIGNALS OF THE FUZZY CONTROLLER (X,Y,Z,PHI,THETA,PSI)



CONTROL INPUT SIGNALS U1,U2,U3,U4 PID V.S. FUZZY PID(NO NOISE)



TRAJECTORY OF THE QUADROTOR IN 2D AND 3D(ADDED NOISE)



CONTROL INPUT SIGNALS U1,U2,U3,U4 PID V.S. FUZZY PID(ADDED NOISE)

V. CONCLUSION

With the understanding of a mathematical representation of a quadrotor, we developed a Conventional PID controller for altitude and attitude control in a quadrotor which was later simulated in Simulink. We have also developed a Fuzzy-PID for position control in a quadrotor which was also later simulated in Simulink. From our simulation results we concluded that in a noise free system the difference in performance between a Fuzzy-PID and Conventional PID is very small due to proper tuning of the gains. Contrary to when noise is added to the systems, the Fuzzy-PID performs much better than the Conventional-PID.

VI. ACKNOWLEDGMENT

THIS WORKED HAS BEEN APPROVED AND SUPPORTED BY DR. HOMAIFAR.

VII. REFERENCES

[1] He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. The Scientific World Journal, 2014.

[2] Heba talla Mohamed Nabil ElKholy, 2014. Dynamic Modeling and Control of a Quadrotor Using Linear and Nonlinear Approaches, Thesis for degree of Master of Science in Robotics, Control and Smart Systems, 2014

[3] Kotarski, Denis, et al. "Control Design for Unmanned Aerial Vehicles with Four Rotors." *Interdisciplinary Description of Complex Systems*, vol. 14, no. 2, 2016, pp. 236–245., doi:10.7906/indecs.14.2.12.

[4] Gautam, Deepak & Ha, Cheolkeun. (2013). Control of a Quadrotor Using a Smart Self-Tuning Fuzzy PID Controller. International Journal of Advanced Robotic Systems. 10. 1. 10.5772/56911.

[5] Benić, Zoran & Piljek, Petar & Kotarski, Denis. (2016). Mathematical Modelling of Unmanned Aerial Vehicles with Four Rotors. Interdisciplinary Description of Complex Systems. 14. 88-100. 10.7906/indecs.14.1.9.

[6] Kuantama, Endrowednes & Vesselenyi, Tiberiu & Dzitac, S & Tarca, Radu. (2017). PID and Fuzzy-PID control model for quadcopter attitude with disturbance parameter.

International Journal of Computers, Communications and Control. 12. 519-532. 10.15837/ijccc.2017.4.2962.