

# *PID Controller Tuning Optimization with Genetic Algorithms for a Quadcopter*

Komal Khuwaja D/O Shoukat Ali Khuwaja  
Mehran , UET. Jamshoro, Pakistan  
[komalali\\_89@yahoo.com](mailto:komalali_89@yahoo.com)

Ioan Constantin Tarca  
University of Oradea, Romania  
[nelut@uoradea.ro](mailto:nelut@uoradea.ro)

Noor-u-Zaman Lighari  
Mehran , UET. Jamshoro, Pakistan  
[nooruzaman@gmail.com](mailto:nooruzaman@gmail.com)

Radu Catalin Tarca  
University of Oradea, Romania  
[rtarca@uoradea.ro](mailto:rtarca@uoradea.ro)

**Abstract**— This paper is focused on the dynamic of mathematical modeling, stability, nonlinear gain control by using Genetic algorithm, utilizing MATLAB tool of a quadcopter. Previously many researchers have been work on several linear controllers such as LQ method; sliding mode and classical PID are used to stabilize the Linear Model. Quadcopter has a nonlinear dynamics and unstable system. In order to maintain their stability, we use nonlinear gain controllers; classical PID controller provides linear gain controller rather than nonlinear gain controller; here we are using modified PID control to improve stability and accuracy. The stability is the state of being resistant to any change. The task is to maintain the quadcopter stability by improving the performance of a PID controller in term of time domain specification. The goal of PID controller design is to determine a set of gains:  $K_p$ ,  $K_i$ , and  $K_d$ , so as to improve the transient response and steady state response of a system as: by reducing the overshoot; by shortening the settling time; by decrease the rise time of the system. Modified PID is the combination of classical PID in addition to Genetic Algorithm. Genetic algorithm consists of three steps: selection, crossover, and mutation. By using Genetic algorithm we correct the behavior of quadcopter.

**Keywords**—quadcopter; dynamic model, Genetic Algorithm, PID controller

## I. INTRODUCTION

Recently, quadcopters have attracted many researchers because they can replace humans in accomplishing dangerous tasks. A quadcopter has four rotor fitted in a cross-like configuration and potential to soar, hover, take off and land [15]. Currently, quadcopter - unmanned aerial vehicles are used not only for the purpose of military missions but also in civilian, research, and commercial domains [17]. It may include inspection of nuclear reactor, fire safety surveillance, inspection of power lines, law enforcement agencies and also investigations in military and agricultural services. The dynamics of the quadcopter are nonlinear, each quadcopter having six degrees of freedom: three rotational and three translational. A conventional helicopter has two rotors, while a quadcopter has four rotors which enable the takeoff, hover and landing; in order to control all six degrees of freedom, four propeller are present [12]. Quadcopter can move both in prismatic as well as in revolute motion. They change torque with respect to each axis so that the inclination angle of

revolute joints would be achieved. It is essential to have the possibility to decrease or increase the thrust on one or more than one propeller/ Changes in the thrust of any of the propeller causes instability in torque around the rotational Z-axes. The essential movements of a quadrotor are achieved by varying the velocities of each of the four rotors, along these lines changing the lifting power. As a consequence, the quadcopter tilts towards the heading of the moderate turning rotor, which empowers speeding up along that course. Subsequently, control of the tilt points and the movement of the quadcopter are firmly associated, so that the approximation of placement (roll and pitch) is critical. The directions of spinning of all the rotors are tuned to adjust the torque. This norm is also employed to yield the anticipated yaw motions. To reach stable flight, it is essential to merge numerous high accuracy sensors with quick and robust control algorithm. Ashfaq Ahmad Mianet (2008) suggested a non-linear model and a nonlinear control algorithm for a 6 DOF quadcopter. Newton-Euler protocol was used for the nonlinear modeling of the quadcopter. A mathematical model was derived from equations of the quadcopter motion and was gained estimated actuation forces by modelling the aerodynamic coefficients and dynamics of electric motors. The main drawback of this research was that it had important overshoot and also time delays in reaching prescribed value. Santos et (2010) presented a quadcopter by using a smart fuzzy controller. A fuzzy logic controller has been designed and applied in order to control the dynamic model of a quadcopter. The desired roll, pitch, yaw, and height are input values. A simulation result shows that the efficiency of this smart controller is satisfactory. The main drawback of this research was that its controller design was too complex. Jun Li (2011) presented his model to examine the dynamic response and PID control algorithm for a quadcopter. This study defines the construction of a quadcopter and elucidates its dynamic model. Apart from it, this study also proposed a controller which aims to control the position and orientation of the 6 DOF model.

Their model shows instability if the values of  $K_p$ ,  $K_i$ , and  $K_d$  are not compatible with each other. Several controllers such as LQ method, PID, Sliding Mode are used to stabilize quadcopter systems. PID controllers, though being efficient, comprise drawbacks of instability due to the incompatibility between the values of  $K_p$ ,  $K_i$ , and  $K_d$  and

requires plant model. PID controller's performance could be dramatically improved if its gains are optimized or nearly optimize.

The paper consists in two parts: mathematical model of a quadcopter and prediction of the PID controller's nonlinear behavior by using Genetic Algorithm (GA). The Aim of the research is to enhance the quadcopter PID controller stability by achieving optimized controller value using GA..For this, the following steps must be taken:

- Modeling of quadcopter.
- Controlling the quadcopter using PID.
- Optimizing the PID parameters using Genetic Algorithm.
- Comparing the conventional PID with GA PID controller.

## II. QUADCOPTER MODEL

### A. Dynamic model

A quadcopter is a 6 DOF system. Let us consider (Fig. 1) x, y, z the quadcopter's center of mass relative to an Earth fixed inertial frame NED. Vector p is the position vector which indicates the distance between Earth frame and body frame;  $\varphi$  (roll angle about x-axis),  $\theta$  (pitch angle about y-axis), and  $\psi$  (yaw angle about z-axis) are the three Euler angles representing the orientation of the quadcopter.

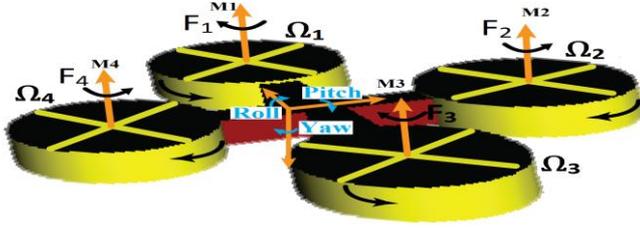


Fig. 1. Quadcopter dynamics

The position and orientation of the quadrotor are given through:

$$p=[x \quad y \quad z]^T \quad (1)$$

$$\eta=[\varphi \quad \theta \quad \psi]^T \quad (2)$$

The rotation R between the body frame and inertial frame represents the orientation of the quadcopter. Assuming the order of rotation to be: roll ( $\varphi$ ), pitch ( $\theta$ ) then yaw ( $\psi$ ), then the rotation matrix R which is derived based on the sequence of principle rotations will be:

$$R=\begin{bmatrix} c\theta c\psi & s\varphi s\theta c\psi-c\varphi s\psi & c\varphi s\theta c\psi+s\varphi s\psi \\ c\theta s\psi & s\varphi s\theta s\psi+c\varphi c\psi & c\varphi s\theta s\psi-s\varphi c\psi \\ -s\theta & s\varphi c\theta & c\varphi c\theta \end{bmatrix} \quad (3)$$

Some states are measure in body frame (e.g., thrust force) while some are measured in an inertial frame (e.g., gravitational forces and quadcopter's position).In order to relate Euler rates (which are measured in inertial frame) and angular body rates  $\omega = [p \ q \ r]^T$ , a transformation is needed as follows:

$$\omega=R_r \cdot \dot{\eta} \quad (4)$$

$$R_r=\begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\varphi & s\varphi c\theta \\ 0 & -s\varphi & c\varphi c\theta \end{bmatrix} \quad (5)$$

Using the Newton-Euler Formalism the rotational equations of motion can be derived as [16]:

$$J \cdot \dot{\omega} + \omega \times J \cdot \omega + M_G = M_B \quad (6)$$

$$J \cdot \dot{\omega} + \omega \times J \cdot \omega + \omega \times [0 \quad 0 \quad J_r \cdot \Omega_r] = M_B \quad (7)$$

$$\Omega_r = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4 \quad (8)$$

where:

J - quadcopter's diagonal inertia matrix

$\omega$  - angular body rates

$M_G$  - gyroscopic moments due to rotor's inertia

$M_B$  - moments acting on the quadcopter in body frame

$J_r$  - rotor's inertia

$\Omega_r$  - rotors' relative speed

$F_i$  - aerodynamic force produced by i<sup>th</sup> rotor

$M_i$  - aerodynamic moment produced by i<sup>th</sup> rotor dynamics

$\rho$  - air density

A - blade area

$C_T, C_D$  - aerodynamic coefficients

r - radius of theblade

$\Omega_i$  - angular velocity of rotor i.

$$J=\begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (9)$$

$$M_B=\begin{bmatrix} L \cdot K_f \cdot (-\Omega_2^2 + \Omega_4^2) \\ L \cdot K_f \cdot (\Omega_1^2 - \Omega_3^2) \\ K_M \cdot (\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \end{bmatrix} \quad (10)$$

$$F_i = \frac{1}{2} \cdot \rho \cdot A \cdot C_T \cdot r^2 \cdot \Omega_i \quad (11)$$

$$M_i = \frac{1}{2} \cdot \rho \cdot A \cdot C_D \cdot r^2 \cdot \Omega_i \quad (12)$$

For the rotational equations of motion result:

$$\begin{bmatrix} I_{xx} \cdot \ddot{\varphi} \\ I_{yy} \cdot \ddot{\theta} \\ I_{zz} \cdot \ddot{\psi} \end{bmatrix} + \begin{bmatrix} \dot{\theta} \cdot I_{zz} \cdot \dot{\psi} - \dot{\psi} \cdot I_{yy} \cdot \dot{\theta} \\ \dot{\psi} \cdot I_{xx} \cdot \dot{\varphi} - \dot{\varphi} \cdot I_{zz} \cdot \dot{\psi} \\ \dot{\varphi} \cdot I_{yy} \cdot \dot{\theta} - \dot{\theta} \cdot I_{xx} \cdot \dot{\varphi} \end{bmatrix} + \begin{bmatrix} \dot{\theta} \cdot J_r \cdot \Omega_r \\ -\dot{\varphi} \cdot J_r \cdot \Omega_r \\ 0 \end{bmatrix} = \begin{bmatrix} LU_2 \\ LU_3 \\ LU_4 \end{bmatrix} \quad (13)$$

And finally:

$$\ddot{\varphi} = \frac{L}{I_{xx}} \cdot U_2 - \frac{J_r}{I_{xx}} \cdot \dot{\theta} \cdot \Omega_r + \frac{I_{yy} \cdot I_{zz}}{I_{xx}} \cdot \dot{\theta} \cdot \dot{\psi} \quad (14)$$

$$\ddot{\theta} = \frac{L}{I_{yy}} \cdot U_3 - \frac{J_r}{I_{yy}} \cdot \dot{\varphi} \cdot \Omega_r + \frac{I_{zz} \cdot I_{xx}}{I_{yy}} \cdot \dot{\varphi} \cdot \dot{\psi} \quad (15)$$

$$\ddot{\psi} = \frac{L}{I_{zz}} \cdot U_4 + \frac{I_{xx} \cdot I_{yy}}{I_{zz}} \cdot \dot{\varphi} \cdot \dot{\theta} \quad (16)$$

Note: U1 is responsible for hovering, U2 is responsible for rolling, U3 is responsible for pitching, and U4 is responsible for yawing.

To obtain the quadcopter movement equations referring to the translation system was used Newton's second law:

$$m \cdot \vec{p} = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} + R \cdot \vec{F}_B \quad (17)$$

$$F_B = \begin{bmatrix} 0 \\ 0 \\ -K_f (\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \end{bmatrix} \quad (18)$$

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} + \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -U_1 \end{bmatrix} \quad (19)$$

where:  $p$  – quadcopter's distance from the inertial frame,  $m$  – quadcopter's mass,  $g$  – gravitational acceleration,  $F$ – non-gravitational forces acting on the quadcopter in the body frame.

$$\ddot{x} = \frac{U_1}{m} (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi) \quad (20)$$

$$\ddot{y} = \frac{U_1}{m} (\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi) \quad (21)$$

$$\ddot{z} = g - \frac{U_1}{m} (\cos\phi \cos\theta) \quad (22)$$

### B. Rotor Dynamics

The DC brushless motor is used in quadcopter model because its provide less friction and high torque, at the steady state the dynamic of DC brushless is same as DC motor shown in Fig. 2. [5]

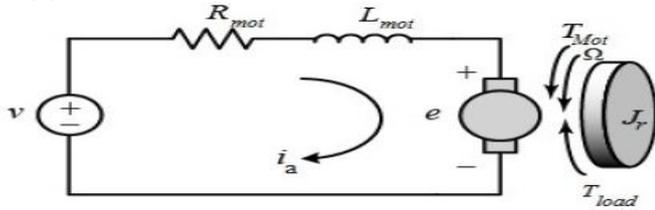


Fig. 2 DC motor dynamic

From this DC dynamic model by using Kirchoff law we derived the equation as following below

$$v = R_{mot} i_a + L_{mot} \frac{di_a}{dt} + K_{mot} \Omega \quad (23)$$

$v$  is the input voltage of a DC motor and  $R_{mot}$  is the resistance and  $L_{mot}$  is the inductance of a motor  $i_a$  is the armature current,  $K_{mot} \Omega$  term describe the generated emf, quadcopter needs small motor and there inductance is very small, so it can be neglected

$$v = R_{mot} i_a + K_{mot} \Omega \quad (24)$$

$$i_a = \frac{v - K_{mot} \Omega}{R_{mot}} \quad (25)$$

And the mechanical part attach to this motor is given by

$$J_r \dot{\Omega}_i = T_{mot} - T_{load} \quad (26)$$

Where  $T_{mot}$  is the torque produce by the motor and is equal to the  $K_e i_a$ ,  $K_e$  is the electric constant and due to the small DC motor it is approximately equal to the  $K_{mot}$ .  $T_{load}$  is the load torque which is the generates by the propeller and is equal to the  $K_M \Omega^2$ , put the value of  $T_{mot}$  and  $T_{load}$  into that equation (26) and simplified we get

$$v = \frac{R_{mot}}{K_{mot}} J_r \Omega_i + K_{mot} \Omega_i + K_M R_{mot} \Omega_i^2 \quad (27)$$

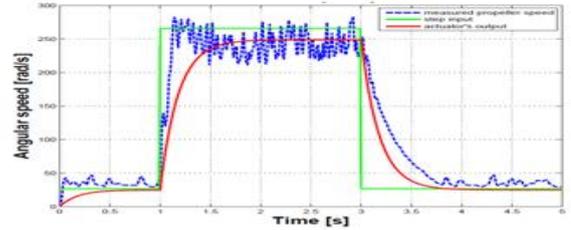


Fig. 3. Angular speed

### C. State Space Model

Using the state space model, one can formulate the state variables of the rotational and translational dynamics as

$$X = [\varphi \ \dot{\varphi} \ \theta \ \dot{\theta} \ \psi \ \dot{\psi} \ z \ \dot{z} \ y \ \dot{y} \ z \ \dot{z}] \quad (28)$$

We can redefinethe state vector as

$$X = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \ x_7 \ x_8 \ x_9 \ x_{10}] \quad (29)$$

Introducing the following constants and state vector, we can express the six second-order differential equations as:

$$a_1 = \frac{I_{yy} - I_{zz}}{I_{xx}}, a_2 = \frac{J_r}{I_{xx}} \quad (30)$$

$$a_3 = \frac{I_{zz} - I_{xx}}{I_{yy}}, a_4 = \frac{J_r}{I_{yy}} \quad (31)$$

$$a_5 = \frac{I_{xx} - I_{yy}}{I_{zz}} \quad (32)$$

$$b_1 = \frac{1}{I_{xx}}, b_2 = \frac{1}{I_{yy}}, b_3 = \frac{1}{I_{zz}} \quad (33)$$

$$\ddot{x} = \frac{-U_1}{m} (\cos x_1 \sin x_5 \sin x_3 + \sin x_1 \sin x_5) \quad (34)$$

$$\ddot{y} = \frac{-U_1}{m} (\cos x_1 \sin x_5 \sin x_3 - \cos x_5 \sin x_1) \quad (35)$$

$$\ddot{z} = g - \frac{U_1}{m} (\cos x_1 \cos x_3) \quad (36)$$

$$\ddot{\varphi} = b_1 U_2 - a_2 x_4 \Omega_r + a_1 x_4 x_6 \quad (37)$$

$$\ddot{\theta} = b_2 U_3 - a_4 x_2 \Omega_r + a_3 x_2 x_6 \quad (38)$$

$$\ddot{\psi} = b_3 U_4 - a_5 x_2 x_4 \quad (39)$$

$$\dot{x}_1 = \dot{\varphi} = x_2 \quad (40)$$

$$\dot{x}_2 = \ddot{\varphi} = b_1 U_2 - a_2 x_4 \Omega_r + a_1 x_4 x_6 \quad (41)$$

$$\dot{x}_3 = \dot{\theta} = x_4 \quad (42)$$

$$\dot{x}_4 = \ddot{\theta} = b_2 U_3 - a_4 x_2 \Omega_r + a_3 x_2 x_6 \quad (43)$$

$$\dot{x}_5 = \dot{\psi} = x_6 \quad (44)$$

$$\dot{x}_6 = \dot{\psi} = b_3 U_4 - a_5 x_2 x_4 \quad (45)$$

$$\dot{x}_7 = \dot{z} = x_8 \quad (46)$$

$$\dot{x}_8 = \dot{z} = g - \frac{U_1}{m} (\cos x_1 \cos x_3) \quad (47)$$

$$\dot{x}_9 = \dot{x} = x_{10} \quad (48)$$

$$\dot{x}_{10} = \dot{x} = \frac{-U_1}{m} (\cos x_1 \cos x_5 \sin x_3 + \sin x_1 \sin x_5) \quad (49)$$

$$\dot{x}_{11} = \dot{y} = x_{12} \quad (50)$$

$$\dot{x}_{12} = \dot{y} = \frac{-U_1}{m} (\cos x_1 \sin x_5 \sin x_3 - \cos x_5 \sin x_1) \quad (51)$$

$$f(X,U) = \begin{bmatrix} x_2 \\ b_1 U_2 - a_2 x_4 \Omega_r + a_1 x_4 x_6 \\ x_4 \\ b_2 U_3 - a_4 x_2 \Omega_r + a_3 x_2 x_6 \\ x_6 \\ b_3 U_4 - a_5 x_2 x_4 \\ x_8 \\ g - \frac{U_1}{m} (\cos x_1 \cos x_3) \\ x_{10} \\ \frac{-U_1}{m} (\cos x_1 \cos x_5 \sin x_3 + \sin x_1 \sin x_5) \\ x_{12} \\ \frac{-U_1}{m} (\cos x_1 \sin x_5 \sin x_3 - \cos x_5 \sin x_1) \end{bmatrix} \quad (52)$$

### III. CONTROL OPTIMIZATION AND SIMULATION RESULT

This block diagram gives a general idea of how the system works, and we are interested in controlling the position or the altitude of a quadcopter. The second block diagram shows the position control of a quadcopter.

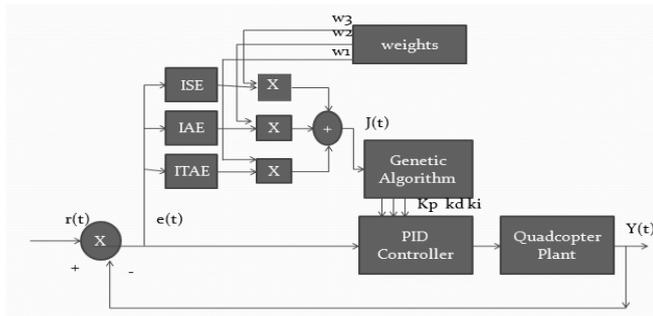


Fig. 4. Block diagram of plant with GA

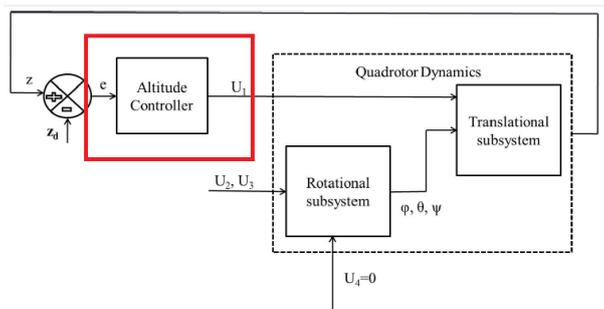


Fig. 5. Block diagram of an altitude control

To control the altitude of a quadcopter we have to go through the following steps:

- to develop the software code and conduct the simulation in the time domain for the quadcopter analysis;
- to adopt the conventional PID gains ( $K_p$ ,  $K_i$ , and  $K_d$ ) with minimum errors between the set value and the actual value;
- to develop GA for the optimization of PID gains using a weighted combination of ISE (Integral square error), IAE (Integral absolute error) and ITAE (Integral time multiple absolute errors) for the improved performance;
- to optimize the PID parameters by using GA and finally by comparing the quadcopter dynamic response between classical PID and GA tuned PID.
- To evaluate the performance of the controller by the simulation results, the quadcopter model and the proposed controller was developed using Matlab/Simulink. In Table I is presented the specifications of the modeled quadcopter.
- The simulation results are given below. Fig. 6 shows the displacement in the z direction of a quadcopter without any controller; the graph shows that the given model is unbounded or we can say unstable and goes toward the infinity. Fig. 7 shows velocity in the z direction and the rate of change in displacement. Fig. 8 and Fig. 9 show the quadcopter model with PID controller having overshoot, high rise time and steady state error due to dynamic model of a system.

TABLE I. QUADCOPTER INITIAL PARAMETERS

Quadcopter Parameter	Values	Quadcopter Parameter	Values
m	0.65(kg)	G	9.81m/s <sup>2</sup>
I <sub>xx</sub>	7.50E-03 kg-m <sup>2</sup>	J <sub>r</sub>	6.00E-05 Kg-m <sup>2</sup>
I <sub>yy</sub>	7.50E-03 Kg-m <sup>2</sup>	K <sub>f</sub>	3.13ee-5 N-s <sup>2</sup>
I <sub>zz</sub>	1.30E-02 Kg-m <sup>2</sup>	K <sub>m</sub>	0.6 Nm-s <sup>2</sup>
l	2.30E-01 m	K <sub>mot</sub>	5.2 mNm/A

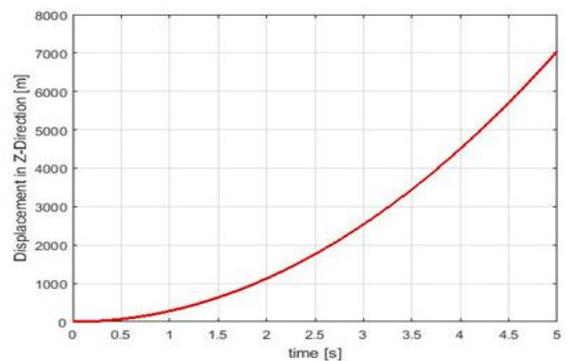


Fig. 6. Displacement of open loop response system

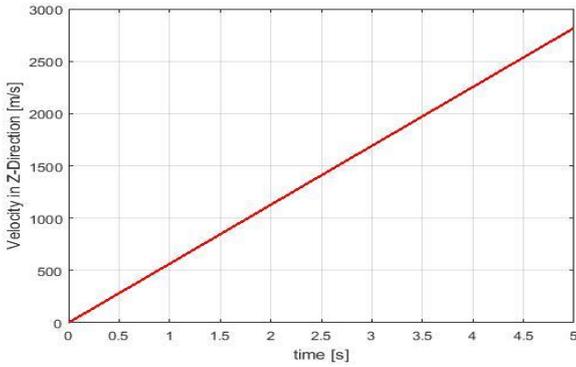


Fig. 7. Velocity of open loop response system

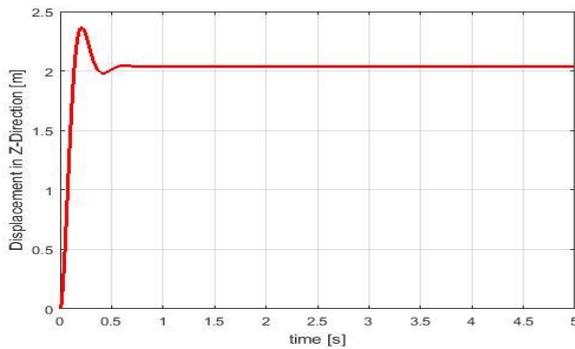


Fig. 8. Displacement response with classical PID

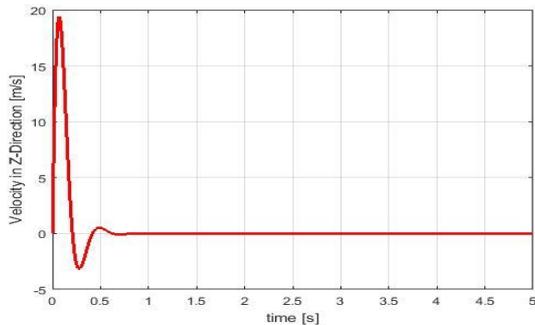


Fig. 9. Velocity response with classical PID

The GA has following advantages:

- easy to understand and implement;
- the algorithm is robust;
- it is a nonlinear process that can a population of points instead of a single solution;
- it is not easily distracted to obtain an optimal local solution instead of a global optimal.
- it does not need information about the system except for the fitness function.

The GA is the powerful optimization technique on tuning application [13]. It has following three steps: reproduction, crossover, and mutation. In reproduction step, we make a selection of the individuals from existing population. In

crossover step, the algorithm creates a new individual from present population. Finally, in mutation step, the algorithm makes modification according to mutation probability.

To obtain the optimal solution the population of a genetic algorithm is evaluated based on fitness function  $f$ . Every individual fitness value either minimizes or maximizes the cost function  $j$  [13]. It can be represented as:

$$f = \frac{1}{j} \quad (53)$$

Fitness value is the product between the cost function and weights. The more the population the better will be the optimal result [3]. let's suppose we are taking 100 percentage of the population, so from 100 percentage of the population we make a cost function to get best fitness value as possible for the result of the minimum cost function. For the cost function we select initial population first, and then we multiply different percentage of weight with available population; we multiply the rest of percentage of weight with new random or guessed value. It is better to take more percentage with given population because we already discarded less bestfitness value from the available population and we take less percentage with new random values. The previous four graphs show the best fitness value. The given system is trying to get more and more bestfitness value for the minimum cost function. The red ones are the worst, the green ones are good, and the blue ones are the best fitness values in which system is more oriented toward the stable state.

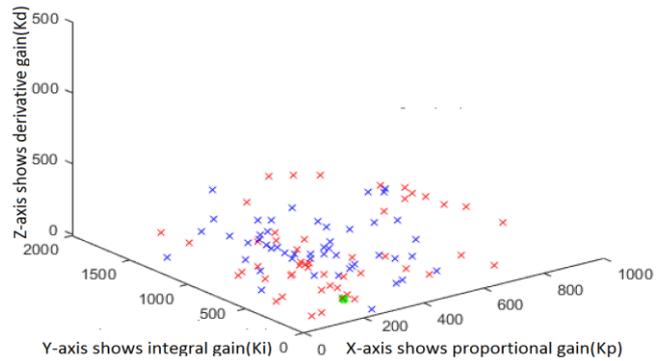


Fig. 10. Initial Selection of fitness value

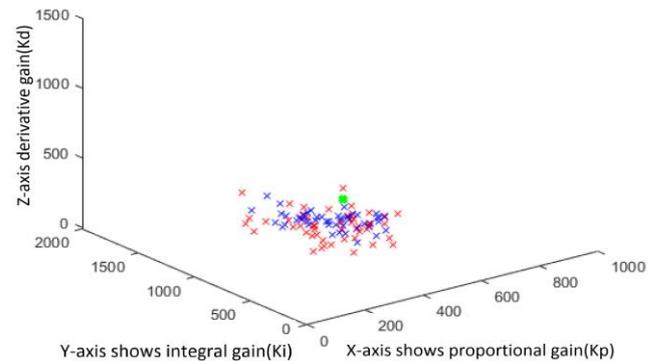


Fig.11. Previousbest fitness value (I)

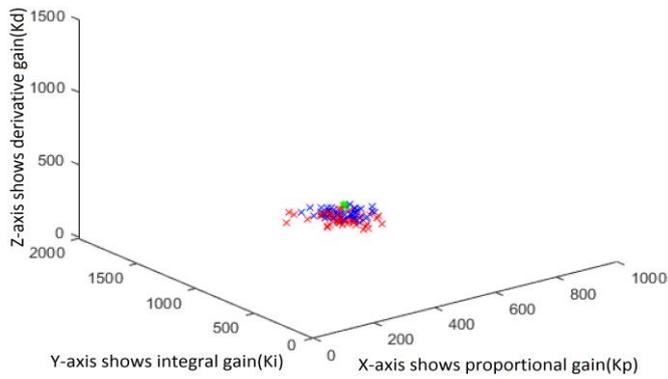


Fig.12. Previous best fitness value II

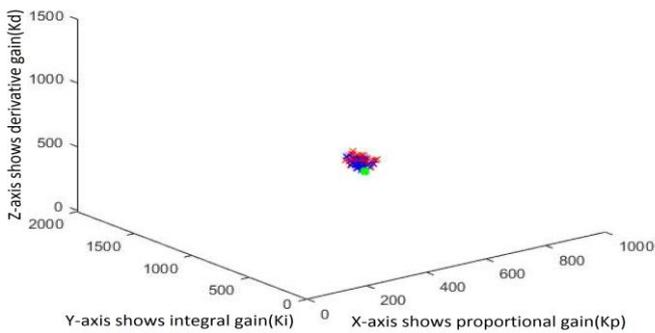


Fig.13. Final best fitness value

The given graph in Fig. 14 and Fig. 15 shows that the best fitness value at which the gain value of  $K_p$ ,  $K_i$  and  $K_d$  make the system more stable as compared to classical PID controller having less rise time and steady state error.

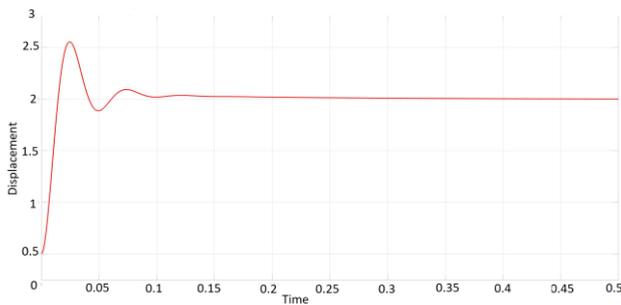


Fig.14. Displacement of system with GA

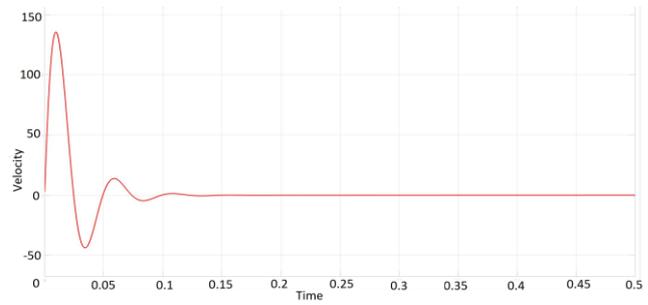


Fig.15. Velocity of system with GA

TABLE.II SIMULATION RESULT PARAMETER

Simulation Result Parameters	Values	Simulation Result Parameters	Values
Rise Time	0.0135s	Best Fitness Value	0.0172
Steady state Error	6.50E-04	Previous Best Fitness Value	0.0172
Possible Peak	0.2801m/s	Cost Function	0.2943

#### IV. CONCLUSION

In this article the dynamic modeling of a quadcopter is presented, obtained through the Newton-Euler formalism. Based on this model, to control the altitude of the quadcopter, a control strategy was developed using GA. The research objective was to develop a PID controller, applying GA to optimize the PID gains, using a weighted combination of ISE (Integral square error), IAE (Integral absolute error) and ITAE (Integral time multiple absolute errors) for the improved performance. Using GA, the gains ( $K_p, K_i, K_d$ ) were tuned through fitness function. The fitness function was our cost function and the minimizing of the cost function was accomplished in order to get more stable controller, to improve the transient and steady state response like settling time, rise time and peak time.

#### Acknowledgment

This work has been funded under the LEADERS - Erasmus Mundus Grant (agreement number 2014-0855/001-001) by European Commission, through the Education, Audio visual and Culture Executive Agency, in the Action Plan 2 for the years 2014-2018, and supported by PNCDI III Program P2 - Transfer of knowledge to the economic operator (Bridge Grant PN-III-P2 2.1 BG-2016-0296) funded by UEFISCDI, Romania.

#### References

- [1] Santos, M., Lopez, V., Morata, F. (2010) Intelligent fuzzy controller of a quadrotor. International Conference on Intelligent Systems and Knowledge Engineering (ISKE), pp.141-146.
- [2] Li, J. and Li, Y. (2011) Dynamic Analysis and PID Control for a Quadrotor. International Conference on Mechatronics and Automation (ICMA), 7-10 August 2011, 573-578.
- [3] Mirzal, Andri, Shinichiro Yoshii, and Masashi Furukawa. "PID parameters optimization by using genetic algorithm." ISTE Journal Of Science And Technology Policy, 2006
- [4] Salih, Atheer L., et al., (2010) "Flight PID controller design for a UAV quadrotor", Scientific Research and Essays, 5.23.

- [5] Habib, Maki K., Wahied Gharieb Ali Abdelaal, and Mohamed Shawky Saad. (2014) "Dynamic modeling and control of a Quadrotor using linear and nonlinear approaches".
- [6] M. Mjahed, (2013) "PID Controller Design using Genetic Algorithm Technique", Proceedings of ICICR, Istanbul, Turkey.
- [7] Mjahed, Mostafa, (2010) "Optimization of classification tasks by using genetic algorithms." *Informatics and Systems (INFOS), 2010 The 7th International Conference on*. IEEE.
- [8] Goldberg, David E., and John H. Holland. "Genetic algorithms and machine learning." *Machine learning* 3.2 (1988): 95-99.
- [9] Bouabdallah, Samir, Andre Noth, and Roland Siegwart. (2004) "PID vs LQ control techniques applied to an indoor micro quadrotor." *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*. Vol. 3. IEEE.
- [10] Mian, Ashfaq Ahmad, and Wang Daobo. "Modeling and backstepping-based nonlinear control strategy for a 6 DOF quadrotor helicopter." *Chinese Journal of Aeronautics* 21.3 (2008): 261-268.
- [11] Mustapa, Muhammad Zaki. (2015). "Altitude Controller Design for Quadcopter UAV", *Jurnal Teknologi*,
- [12] Lower, Michał, and Wojciech Tarnawski. (2015) "Quadrotor navigation using the PID and neural network controller", *Theory and Engineering of Complex Systems and Dependability*. Springer International Publishing.. 265-274.
- [13] Jaen-Cuellar, Arturo Y., et al. (2013) "PID-controller tuning optimization with genetic algorithms in servo systems." *International Journal of Advanced Robotic Systems* 10
- [14] Kuantama, E., Vesselenyi, T., Dzitac, S., & Tarca, R. (2017). PID and Fuzzy-PID Control Model for Quadcopter Attitude with Disturbance Parameter. *International Journal of Computers Communications & Control*, 12(4), 519-532.
- [15] Syam, Rafiuddin, (2010) "Simulation and Experimental Works of Quadcopter Model for Simple Maneuver." *The 1<sup>st</sup> International Symposium On Smart Material And Mechatronics*.
- [16] Deepak Gautam and Cheolkeun Ha, (2013), Control of a Quadrotor Using a Smart Self-Tuning Fuzzy PID Controller, *International Journal of Advanced Robotic Systems*, vol. 10, pp.1-9.
- [17] Safaei Ali and Mahyuddin Muhammad Nasiruddin, (2016), Lyapunov-based Nonlinear Controller for Quadrotor Position and Attitude Tracking with GA Optimization, *Industrial Electronics and Applications Conference (IEACon)*, IEEE, pp. 342-347.