# Mathematical Modeling and Designing PID Controller for a Quadrotor and Optimizing its Step Response by Genetic Algorithm

D. Sabzevari[1], S. M. Kargar[2], S. M. A. Zanjani[3]
1- Department of Electrical Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran.
Email[1]: sabzevari@sel.iaun.ac.ir
2,3- Department of Electrical Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran.
Email[2]: Kargar@pel.iaun.ac.ir , Email[3]:Sma_zanjani@pel.iaun.ac.ir

Received: March 2016 Revised: May 2016 Accepted: June 2016

**ABSTRACT:**
In order to analyze the mathematical modeling and PID controller performance of a quadrotor, this paper firstly, describes the quadrotor flight dynamics according to "Newton-Euler laws", then equations of motion linearized and transfer functions for 6 degrees of freedom obtained in state space domain. Classic PID controller based on "Ziegler-Nichols method" is designed and implemented on the system. In order to have better performance, Genetic Algorithm based on step response optimization is used to optimize PID controller performance and compared with classic method. Finally, step responses comparison for each transfer function show that Genetic Algorithm with PID control synthesis better efficiency than the classic PID controller.

**KEYWORDS:** Quadrotor, PID controller, Ziegler-Nichols, 6 degrees of freedom, Genetic Algorithm

## 1. INTRODUCTION

A quadrotor is a kind of multi-rotor aircraft which can achieve Vertical Take-Off and Landing (VTOL). The flight attitude control of the quadrotor can be achieved only by adjusting the speed of the four rotors. Smaller dimension, less weight, more flexibility, less noise and easier maintenance, make quadrotors more popular than other vertical flight robots like Helicopters, Hexa-rotors, etc. As in [1] and [2], Unmanned Aerial Vehicles (UAVs) have the potential for full-filling many civil and military applications including surveillance, intervention in hostile environments, air pollution monitoring, and area mapping. In the quadrotor, there are four rotors with fixed angles which represent four input forces that are basically the thrust generated by each propeller.

According to [3], [4] and [5], Control and stability of the robot are defined by changing the motor`s speed in processors. In this paper, modeled quadrotor is supposed to 6 degrees of freedom in order to obtain angle and position of the robot synchronously. The control input is a signal called error that is the difference between desire input voltage and a voltage that is received from different sensors such as Accelerometer, GPS, and Laser Scanners as a feedback signal.

One purpose of control is minimizing error signal by designing and tuning the controller coefficients. In this paper, in order to design the optimum controller, firstly, nonlinear and unstable dynamics are studied and described in time domain, then, transfer functions are determined in state space domain. Nonlinear equations are linearized in equilibrium point, then according to [7] and [8], the PID controller is designed and regulated. In purpose of stability criteria for this paper, as in [10], Genetic Algorithm based on step response optimization, is used and the optimized PID controller is compared with classic method and results are simulated in Matlab.

## 2. ROBOT STRUCTURE AND ITS MOVEMENT

Quadrotor's frame is included in 2 type, "X" configuration (Fig. 1) and "plus" configuration (Fig.2) that have no such difference on stability aspects. So in this paper, the plus configuration is studied.

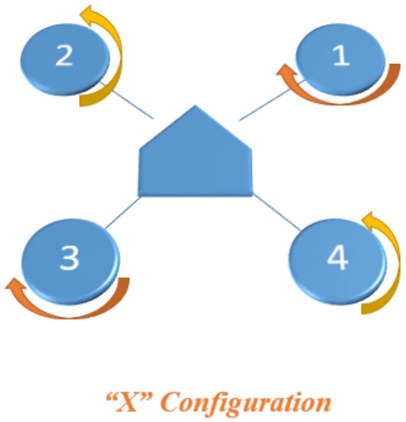


**Fig. 1**. "X" configuration

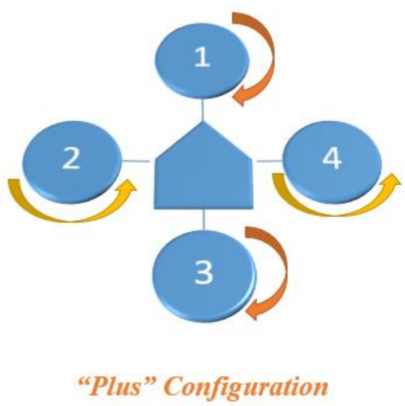


**Fig. 2**. "Plus" configuration

This robot as shown in (Fig. 3), has three straight movements and three rotating movements. To considering robot movements, two coordinate axes are defined:

I. **Inertial frame:** This coordinate axis is assumed constant on earth. The axes are shown respectively with X, Y, Z.

XY plate is set on horizon and the Z axis is specified into down side by right hand rule.

II. **Body frame:** This coordinate axis is supposed on robot`s body and rotates along with that and the center is matched on center of robot`s body. Axes are shown respectively with X`, Y`, Z`.

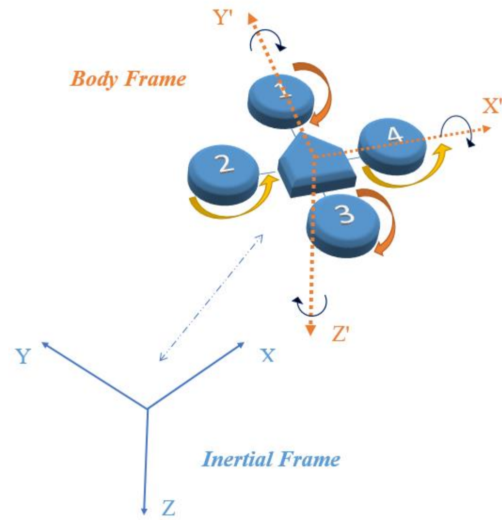


**Fig. 3**. Inertial frame and Body frame

Quadrotor has three rotating movements around its coordinate axes, that rotating around X axis is called 'Roll angular' and rotating around Y axis is called 'Pitch angular', rotating around Z axis is called 'Yaw angular'. Therefore, as in "Fig. 4," they are shown respectively with φ, θ and ψ.

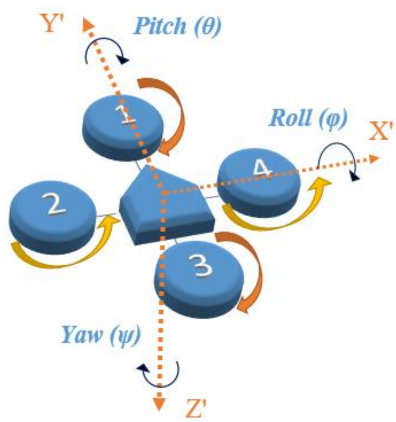


**Fig. 4**. Roll, Pitch and Yaw angles

Within changing the motor RPM, thrust force and lift force will change so that cause the robot movements.
The motors rotation as shown in Fig. 4, are as follows that, motor 1 and 3 rotate clockwise and motor 2 and 4 rotate counter-clockwise. Motor performance in rotating around the axes is following as below:

a. **Rotating around X axis:** In roll motion, the torque of motor 2 is increased and the torque of motor 4 (in order to equivalent torques of motor 3 and 1 with the total torque of motor 2 and 4) will be decreased.

b. **Rotating around Y axis:** In pitch motion, the torque of motor1 is increased and the torque of motor 3 (in order to equivalent the total torques of motor 2 and 4 and the total torques of motor 1 and 3) will be decreased.
c. **Rotating around Z axis:** For making Yaw motion, the speed of motor 3 and 1 must be increased in the same time and also, the speed of motor 2 and 4 must be decreased.
d. **Increasing or decreasing the height:** If speed of all motors, in the same time and magnitude, increase or decrease, the height will be increased or decreased.

## 3. EQUATIONS OF ROTATIONAL MOTION

As in [1], for mathematical modeling, Newton-Euler formula is used as below:

$$\vec{F} = \frac{d(m\vec{V})}{dt} \quad (1)$$

$$\vec{M} = \frac{d(\vec{H})}{dt} \quad (2)$$

$$\frac{d}{dt}\iiint \vec{r} * \frac{d(\vec{r})}{dt}\rho * d\forall + \frac{d}{dt} * (\vec{h}) = \text{Moments} \quad (3)$$

$(\vec{M})$ is consequence of entered torques, $(\vec{V})$ is linear velocity of robot, $(\vec{H})$ is angular moment of robot and (m) is shown the mass. Equation (3) is defined for linear and angular moment according to Newton second law. $(\vec{h})$ is angular moment of rotor, (F) is consequent of external forces, $(\rho)$ is bulk density, $(\vec{r})$ is distance of center of robot to integrating level and (Moments) show the moment of aerodynamic and motor forces.
"Equation (4)" show the moment for a rigid body:

$$(\vec{h}) = \sum_{i=1}^{n} I_{R_i} * \vec{\omega}_{R_i} \quad (4)$$

That $(I_{R_i})$ is inertia moment and $(\vec{\omega}_{R_i})$ is angular velocity. $(\vec{h})$ in 3D is define in (5):

$$(\vec{h}) = \vec{i}h_x + \vec{j}h_y + \vec{k}h_z \quad (5)$$

With replace $(\vec{h})$ in (3), general form for equation of rotational motion is obtained this::

$$\begin{cases} 1)\ I_{xx}\dot{P} - I_{xz}\dot{R} - I_{xz}PQ + (I_{zz} - I_{yy})RQ \\ \quad +Qh_z - Rh_y + \dot{h}_x = L_A + L_T \\ 2)\ I_{yy}\dot{Q} + (I_{xx} - I_{zz})PR + I_{xz}(P^2 - R^2) \\ \quad +Rh_x - Ph_z + \dot{h}_y = M_A + M_T \\ 3)\ I_{zz}\dot{R} - I_{xz}\dot{P} + (I_{yy} - I_{xx})PQ + I_{xz}QR \\ \quad +Ph_y - Qh_x + \dot{h}_z = N_A + N_T \end{cases} \quad (6)$$

As in (6), $I_{xx}$, $I_{yy}$, $I_{zz}$ and $I_{xz}$ are inertia moment around the coordinate axes and P, Q and R, are respectively angular velocity of X, Y, Z axes in inertial frame. $L_A$, $M_A$ and $N_A$ are moment of aerodynamic forces. $L_T$, $M_T$ and $N_T$ are moments of rotor forces.

## 4. TRANSFORMATION MATRIX AND EULER ANGLES

As in [1], [2] and [4], the horizontal coordinate axis could match on the body frame with three continuous rotate that the rotational matrix is gained in (7). Because the horizontal coordinate axis is in the same direction as an inertial coordinate axis so this rotational matrix is true for converting from inertial frame to body frame too. In order to gain the angular velocity around body axes, the rotational matrix as given below must be used:

$$R_r = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\varphi & \cos\theta\sin\varphi \\ 0 & -\sin\varphi & \cos\theta\cos\varphi \end{bmatrix} \quad (7)$$

$\theta, \varphi$ and $\psi$ are Euler angular. Now Equation (7), is used for getting the relation between Euler angular rate $\dot{\theta}, \dot{\varphi}$ and $\dot{\psi}$ and angular velocity around inertial coordinate axes. Generally, the Euler angles rate are not perpendicular on each other and it is necessary that every angular velocity was imagined on body coordinate axes and with sum of all these converts on body coordinate axes, the angular velocity will be gained, so as in (8), the relation between angular velocity around the inertial coordinate axes and Euler angles rate is defined:

$$\begin{bmatrix} P \\ Q \\ R \end{bmatrix} = R_r * \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \Rightarrow \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = R_r^{-1} * \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \quad (8)$$

$$\Rightarrow \begin{cases} \dot{\varphi} = P + Q\sin\varphi * \tan\theta + R\cos\varphi * \tan\theta \\ \dot{\theta} = Q\cos\varphi - R\sin\varphi \\ \dot{\psi} = (Q\sin\varphi + R\cos\varphi) * \sec\theta \end{cases} \quad (9)$$

As in [4], following assumptions are considered:
i. The structure is supposed rigid.
ii. The structure is supposed symmetrical.

So:

$$I_{zx} = I_{xz} = I_{xy} = I_{yx} = 0 \quad (10)$$

$$h_x = h_y = 0 \Rightarrow (\vec{h}) = \vec{k}h_z \quad (11)$$

With considering above assumptions and (8), (9), equation (6), is rewrite in (12).

$$\begin{cases} 1)\ I_{xx}\dot{P} + (I_{zz} - I_{yy})RQ + QI_R\sum_{i=1}^{4}\vec{\omega}_{R_i} = L_A + L_T \\ 2)\ I_{yy}\dot{Q} + (I_{xx} - I_{zz})PR - PI_R\sum_{i=1}^{4}\vec{\omega}_{R_i} = M_A + M_T \\ 3)\ I_{zz}\dot{R} + (I_{yy} - I_{xx})PQ + I_R\sum_{i=1}^{4}\vec{\omega}_{R_i} = N_A + N_T \end{cases} \quad (12)$$

Equation (12) is final equation of rotational motion in time domain.

## 5. EQUATIONS OF TRANSLATIONAL MOTION

According to [2], [3] and [4], to studying the Equations of translational motion in inertial frame, the Rotary matrix ($R_t$) is used as in (13).

$$R_t = \begin{bmatrix} \cos\theta\cos\psi & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \sin\psi\sin\phi + \cos\psi\cos\phi\sin\theta \\ \cos\theta\sin\psi & \cos\psi\cos\phi + \sin\phi\sin\theta\sin\psi & \cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \quad (13)$$

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} m\ddot{X} \\ m\ddot{Y} \\ m\ddot{Z} \end{bmatrix} = R_t * \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} - \begin{bmatrix} F_{D_x} \\ F_{D_y} \\ F_{D_z} \end{bmatrix} \quad (14)$$

That, f is translational force in body frame and F is translational force in inertial frame.

$$\begin{cases} 1)\quad F_x = f_x\cos\theta\cos\psi + \\ f_y(\cos\psi\sin\theta\sin\varphi - \cos\theta\sin\psi) \\ +f_z(\sin\varphi\sin\psi + \cos\varphi\cos\psi\sin\theta) - F_{D_x} \\ 2)\quad F_y = f_x\cos\theta\sin\psi + \\ f_y(\cos\theta\cos\psi + \cos\theta\sin\varphi\sin\psi) \\ +f_z(\cos\theta\sin\theta\sin\psi - \cos\psi\sin\varphi) - F_{D_y} \\ 3)\quad F_z = -f_x\sin\theta + \\ f_y(\cos\theta\sin\varphi) + f_z(\cos\varphi\cos\varphi) - F_{D_z} \end{cases} \quad (15)$$

$F_{D_z}$, $F_{D_y}$ and $F_{D_x}$ are drag forces in order to X, Y and Z axes.

$$\begin{cases} 1)\ m\ddot{X} = -(\sin\varphi\sin\psi + \cos\varphi\cos\psi\sin\theta) \\ \quad * \sum_{i=1}^{4}T_i - \frac{1}{2}C_xA_c\rho\dot{X}|\dot{X}| - \sum_{i=1}^{4}H_{xi} \\ 2)\ m\ddot{Y} = -(\cos\theta\sin\theta\sin\psi - \cos\psi\sin\varphi) \\ \quad * \sum_{i=1}^{4}T_i - \frac{1}{2}C_yA_c\rho\dot{Y}|\dot{Y}| - \sum_{i=1}^{4}H_{yi} \\ 3)\ m\ddot{Z} = -(\cos\varphi\cos\varphi)\sum_{i=1}^{4}T_i - \frac{1}{2}C_zA_c\rho\dot{Z}|\dot{Z}| \end{cases} \quad (16)$$

Equation (16), is related to translational motion in time domain, that $T_i$ is thrust force, A and C are motor's dynamic coefficients that will obtain in (19). Motor's dynamic relation is shown in (17).

$$\begin{cases} \dot{\omega}_m = \frac{1}{\tau}\omega_m - \frac{d}{\eta r^3 J_t}\omega_m^2 + \frac{1}{k_m\tau}u \\ \frac{1}{\tau} = \frac{k_m^2}{R_{mot}J_t} \end{cases} \quad (17)$$

Equation (17) is linearized at$_0$ point, as a result new equation of motor is obtained in (18).

$$\dot{\omega}_m = -A\omega_m + Bu + C \quad (18)$$

That A, B and C are attained:

$$\begin{cases} A = \left(\frac{1}{\tau} + \frac{2d\omega_0}{\eta r^3 J_t}\right) \\ B = \left(\frac{1}{k_m\tau}\right) \\ C = \left(\frac{d\omega_0^2}{\eta r^3 J_t}\right) \end{cases} \quad (19)$$

The motor's parameters are defining in "Table (1)".

**Table 1**. Motor specifications

| Parameter | Definition |
|---|---|
| $\tau$ | Torque constant |
| $\eta$ | Gearbox efficiency |
| r | Gearbox reduction ratio |
| $J_t$ | Total rotor inertia is seen by the motor |
| $K_m$ | Motor torque constant |
| $R_{mot}$ | Rotor internal resistance |
| u | Motor input voltage |
| $\omega_m$ | Motor angular rate |
| d | Drag factor |

## 6. EQUATIONS OF STATE SPACE

According to [5] and [6], the obtained dynamic equations will be rewrite in state space.
The state space equation is defining as below:

$$\dot{X} = f(X, U) \quad (20)$$

That X is state variable and U is control input.

$$X = [\varphi\ \dot{\varphi}\ \theta\ \dot{\theta}\ \psi\ \dot{\psi}\ z\ \dot{z}\ x\ \dot{x}\ y\ \dot{y}]^T \quad (21)$$

The state space inputs and variables are defined in (22) and (23):

$$\begin{array}{ll} x1 = \varphi & ,\ x7 = z \\ x2 = \dot{x1} = \dot{\varphi} & ,\ x8 = \dot{x7} = \dot{z} \\ x3 = \theta & ,\ x9 = x \\ x4 = \dot{x3} = \dot{\theta} & ,\ x10 = \dot{x9} = \dot{x} \\ x5 = \psi & ,\ x11 = y \\ x6 = \dot{x5} = \dot{\psi} & ,\ x12 = \dot{x11} = \dot{y} \end{array} \quad (22)$$

$$U = [U_1\ U_2\ U_3\ U_4]^T \quad (23)$$

$U_1, U_2, U_3$ and $U_4$ are control inputs that is given by equation (24).

$$\begin{cases} U_1 = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ U_2 = b(-\Omega_2^2 + \Omega_4^2) \\ U_3 = b(\Omega_1^2 - \Omega_3^2) \\ U_4 = d(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2) \end{cases} \tag{24}$$

The parameters of above equation are defined in Table (2).

**Table 2.** Definitions of control input parameters

| Parameter | Definition |
|---|---|
| $U_1$ | Total thrust force of all motors |
| $U_2$ | Difference thrust force between motor 2,4 for roll rotation |
| $U_3$ | Difference thrust force between motor 1,3 for pitch rotation |
| $U_4$ | Total thrust forces for yaw rotation |
| $\Omega$ | Propeller angular rate |
| b | Trust factor |

By rewriting the equations, according to control inputs (U), the state space matrix is obtained as shown in (25).

$$f(X.U) = \begin{bmatrix} \dot{\varphi} \\ \dot{\psi}\dot{\varphi}a_1 - \theta\dot{\Omega}_r a_2 + b_1 U_2 \\ \dot{\theta} \\ \dot{\psi}\dot{\varphi}a_3 + \dot{\varphi}\dot{\Omega}_r a_4 + b_2 U_3 \\ \dot{\psi} \\ \dot{\theta}\dot{\varphi}a_5 + b_3 U_4 \\ \dot{z} \\ g - \frac{(\cos\theta\cos\varphi)U_1}{m} \\ \dot{x} \\ U_x \frac{U_1}{m} \\ \dot{y} \\ U_y \frac{U_1}{m} \end{bmatrix} \tag{25}$$

$a_1, a_2, a_3, a_4, a_5, b_1, b_2, b_3$ and $\Omega_r$ are defined as below.

$$\begin{aligned} a_1 &= \left(\frac{I_{yy} - I_{zz}}{I_{xx}}\right) \quad &, b_1 = \left(\frac{L}{I_{xx}}\right) \\ a_2 &= \left(\frac{I_R}{I_{xx}}\right) \quad &, b_2 = \left(\frac{L}{I_{yy}}\right) \\ a_3 &= \left(\frac{I_{zz} - I_{xx}}{I_{yy}}\right) \quad &, b_3 = \left(\frac{L}{I_{zz}}\right) \\ a_4 &= \left(\frac{I_R}{I_{yy}}\right) \quad &, \Omega_r = \sum_{i=1}^{4} \Omega_i \\ a_5 &= \left(\frac{I_{xx} - I_{yy}}{I_{zz}}\right) \end{aligned} \tag{26}$$

$$\begin{cases} U_x = \sin\psi\sin\varphi + \cos\varphi\cos\psi\sin\theta \\ U_y = -\cos\psi\sin\varphi + \sin\psi\sin\theta\cos\varphi \end{cases}$$

L is horizontal distance of propeller center to center of gravity. Equation (27), shows the general nonlinear equations form of rotational and translational motion in state space, however in nonlinear equations of hover state, the cross coupling and gyroscopic effects can be omitted.

$$\begin{cases} \dot{\theta}\dot{\varphi} = \dot{\psi}\dot{\varphi} = \dot{\theta}\dot{\psi} = 0 \\ I_R \Omega_r \dot{\theta} = I_R \dot{\varphi}\dot{\Omega}_r = 0 \end{cases} \tag{28}$$

So within hover state assumption, equations are simplified as below:

$$\begin{cases} I_{xx}\ddot{\varphi} = LU_2 \\ I_{yy}\ddot{\theta} = LU_3 \\ I_{zz}\ddot{\psi} = U_4 \end{cases} \tag{29}$$

In order to achieve the transfer functions, rotor dynamics and the model are rewritten in Laplace transform.

$$\begin{cases} \varphi(s) = \frac{B^2 bl}{s^2(s+A)^2 I_{xx}}\left(u_4^2(s) - u_2^2(s)\right) \\ \theta = \frac{B^2 bl}{s^2(s+A)^2 I_{yy}}\left(u_3^2(s) - u_1^2(s)\right) \\ \psi(s) = \frac{B^2 d}{s^2(s+A)^2 I_{zz}}(-1)^{i+1}\sum_{i=1}^{4} u_i^2(s) \end{cases} \tag{30}$$

As in [4], A and B are the coefficients of the linearized rotor dynamics. C is neglected because of its small value compared to B. By replacing the control inputs (U) instead of the motor inputs (u), 'Equations (30)' for attitude control change to (31).

$$\begin{cases} \varphi(s) = \frac{A^2 l}{s^2(s+A)^2 I_{xx}} U_2 \\ \theta(s) = \frac{A^2 l}{s^2(s+A)^2 I_{yy}} U_3 \\ \psi(s) = \frac{A^2}{s^2(s+A)^2 I_{zz}} U_4 \end{cases} \tag{31}$$

According to Equation (16), the position control for quadrotor is obtained as (32).

$$\begin{cases} \frac{X(s)}{\theta(s)} = -g\frac{1}{s^2} \\ \frac{Y(s)}{\varphi(s)} = g\frac{1}{s^2} \\ \frac{Z(s)}{T_n(s)} = \frac{Z(s)}{mg - U_1} = \frac{1}{s^2} \end{cases} \tag{32}$$

## 7. DESIGNING THE PID CONTROLER

In order to control the quadrotor, some parameters make disturbance on system, such as: nonlinear and unstable effects of blades, blades flexibility, ground effect, battery discharging, unknown aerodynamics and etc.

Consequently, as these uncertainties and nonlinear behavior of system, linear controls have good performance just around the equilibrium point. If disturbances become so high, the linear control is not suitable else, so the modern methods will used like optimal control, nonlinear control, adaptive control or etc.

Generally, the robot control block diagram is shown as figure (5):

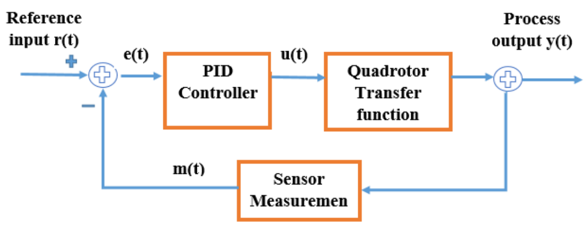


**Fig. 5.** Robot control block diagram

e(t) is error signal and define as in (33).

$$e(t) = r(t) - m(t) \tag{33}$$

m(t) is sensor feedback. According to [7] and [8], the standard form of PID controller shown in (34):

$$C(t) = K_p\left(e(t) + \frac{1}{T_i}\int_0^t e(\tau)d\tau + T_d\frac{de(t)}{dt}\right) \tag{34}$$

$K_p$, $T_i$ and $T_d$ are PID controller coefficients. To eliminate constant disturbance and minimize the steady state error, PID controller is implemented to track the step input precisely and eliminate step disturbance too. According to Table (3), $K_p$, $T_i$ and $T_d$ will be obtained from Ziegler-Nichols method.

**Table 3.** Ziegler-Nichols coefficients table

| Type | $K_p$ | $T_i$ | $T_d$ |
|---|---|---|---|
| **P** | 0.5 $K_u$ | – | – |
| **PI** | 0.45 $K_u$ | $(1/2)K_p/T_u$ | – |
| **PID** | 0.6 $K_u$ | 2 $K_p/T_u$ | $K_pT_u/8$ |

As in [8], $K_u$ will be determined by approaching it to unstable boundary and $T_u$ is oscillations period when system approaching to unstable boundary.

Finally, according to Ziegler-Nichols method, the Table (4) shows the PID controller coefficients which obtained for each quadrotor's transfer function separately as in (31) and (32).

**Table 4.** PID controller coefficients for Transfer functions

| Coefficients / Transfer functions | Kp | Ti | Td |
|---|---|---|---|
| **Roll or Pitch** | 2.52 | 3.93 | 4.02 |
| **Yaw** | 1.2 | 1.57 | 1.6 |
| **X or Y** | 2.1 | 1.73 | 5.43 |
| **Z** | 1.1 | 1.58 | 1.63 |

As in [7], to have desired step response, such as: less than: 5 second settling time, 6% overshoot, 1.5 second rise time and 2% steady state error, in this paper Genetic Algorithm is used to achieve above purposes and better performance.

## 8. GENETIC ALGORITHM

According to [10] and [11], in a Genetic Algorithm (GA), a population of candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties (its chromosomes or genotype) which can be mutated and altered. Genetic Algorithm uses probable rules that named fitness functions in order to produce better and new generation, so this algorithm is supposed to be a model for optimizing a lot of issues. We defined a fitness function based on generation, that mutation, population, the optimization issue will be evaluated. According to [12], 'Equation (35)' is defined as bellow to minimize the cost function (Z).

$$Z = w_1M_p + w_2T_s + w_3T_r \tag{35}$$

That $M_P$ is overshoot percent, $T_S$ is settling time, according to stability analysis, $T_r$ is rise time and $w_1$, $w_2$ and $w_3$ are cost function weights that by weights variation, the stability criteria will be changed. According to Equation (35), 100 iterations and in order to crossover implementation the whole research area is used. The algorithm stop working after 150 generations. As in [13], after defining parameters for GA, for example, number of chromosomes in the population, number of generations, selection and mutation rates, the first generation is randomly generated and algorithm starts a loop.

In each iteration, after decoding the chromosomes and obtaining $K_p$, $T_i$, and $T_d$ coefficients, these numbers are sent to the model to tune the controller. Then the system is run with predefined initial states and its response to step inputs that is feedback to the algorithm, figure (6) show the block diagram of PID controller and GA combination.

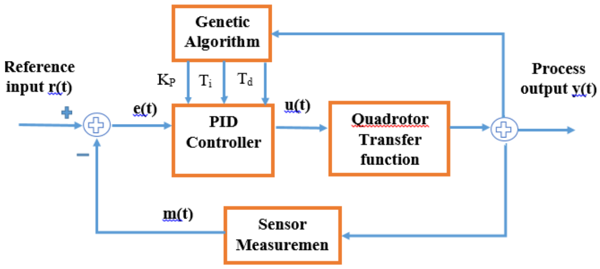


**Fig. 6.** Block diagram of PID controller and GA combination

The optimized PID coefficients will be achieved to use in controller. Table (5) shows the optimized PID controller coefficients.

The comparison between closed loop step response in classic and optimized PID controller with 6 degree of freedom and for each transfer function is shown in figure (7) to figure (10).

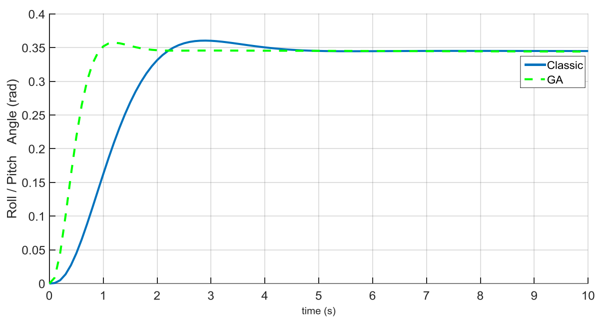


**Fig. 7**. Step response for Pitch or Roll angles

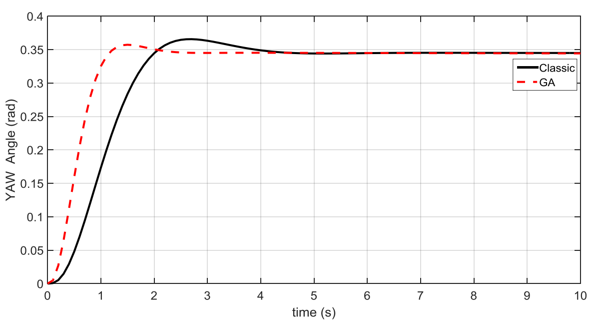


**Fig. 8.** Step response for Yaw angle

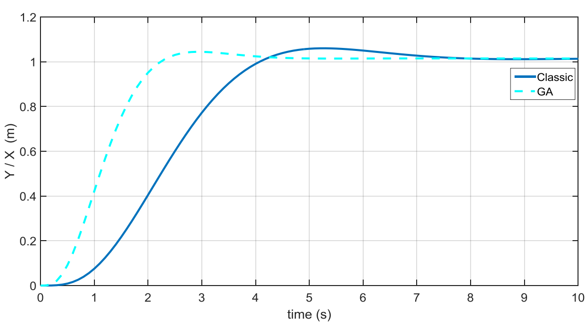


**Fig. 9**. Step response for X or Y positions

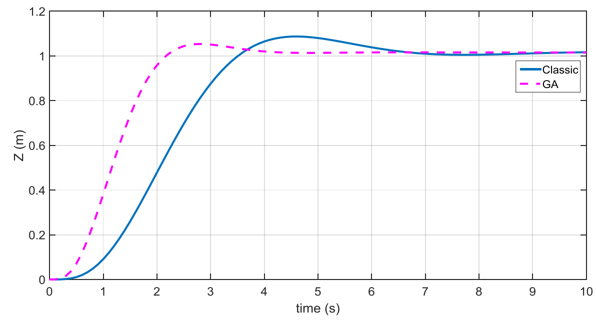


**Fig. 10**. Step response for Z position

The result shows that both controllers can stabilize system and achieve to the desired set points, also Table (5), presents the step response specification comparison of two method, decreasing nearly 2% overshoot, almost 1 second rise time and about 2 second settling time, show that Genetic Algorithm with PID controller synthesis has better performance than the classic PID controller based on Ziegler-Nichols method.

**Table 5.** Step response specifications of classical PID compared with optimized PID

| **step response specifications** / **Transfer Functions** | **Classic PID** | **Optimized PID** |
|---|---|---|
| **Roll or Pitch** | Tr=1.27(sec)<br>Ts=4.4(sec)<br>Mp=6.61(%) | Tr=0.59(sec)<br>Ts=2.79(sec)<br>Mp=5.13(%) |
| **Yaw** | Tr=1.27(sec)<br>Ts=4.4(sec)<br>Mp=6.61(%) | Tr=0.51(sec)<br>Ts=2.16(sec)<br>Mp=4.26(%) |
| **X or Y** | Tr=2.34(sec)<br>Ts=6.97(sec)<br>Mp=5.7(%) | Tr=1.33(sec)<br>Ts=4.03(sec)<br>Mp=4.5(%) |
| **Z** | Tr=2.16(sec)<br>Ts=6.95(sec)<br>Mp=8.29(%) | Tr=1.56(sec)<br>Ts=4.47(sec)<br>Mp=5.24(%) |

## 9. CONCLUSION

In this paper, equations of a quadrotor with 6 degree of freedom have been studied. To control the robot, translational and rotational motion equations are obtained and linearized around the criteria point.

The transfer functions are achieved in Laplace domain and based on Ziegler-Nichols method the PID controller is designed and implemented. To optimize step response and better performance, Genetic Algorithm is implemented. Consequently, optimized PID controller coefficients gained and step response specification compares with classic PID controller as shown in Table (5), which represents Genetic Algorithm with PID controller synthesis is better controller for this system than the classic PID controller.

## 10. ACKNOWLEDGMENT

We would like to show our gratitude to the, Ms. Farnaz Forootan, for giving us a good guidance for edition of this paper.

## REFERENCES

[1] Etkin, B. L. D. Reid, **"Dynamics of flight: stability and control",** *Wiley New York*, 1996.

[2] Schmidt, M. D., "**Simulation and control of a quadrotor unmanned aerial vehicle"**, 2011.

[3] Dikmen, İ., et al., **"Attitude control of a quadrotor",** Recent Advances in Space Technologies, 2009, RAST'09. 4*th International Conference on, IEEE*, 2009.

[4] Bouabdallah, S., **"Design and control of quadrotors with application to autonomous flying"**, *Ecole Polytechnique Federale de Lausanne*, 2007.

[5] Khatoon, S., et al., **"Dynamic modeling and stabilization of quadrotor using PID controller"**, *Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on, IEEE*, 2014.

[6] Salih, A. L., et al., **"Modelling and PID controller design for a quadrotor unmanned air vehicle"**, *Automation Quality and Testing Robotics (AQTR), 2010 IEEE International Conference on*, IEEE, 2010.

[7] Fernandes, S. I., **"Guidance and trajectory following of an autonomous vision-guided micro quadrotor",** *Lisbon, Portugal: Universidade Tecnica de Lisboa*, 2011.

[8] Åström, K., T. Hägglund, **"Revisiting the Ziegler–Nichols step response method for PID control"**, *Journal of process control*, Vol. 14, No. 6, pp. 635-650, 2004.

[9] Åström, K. J., T. Hägglund, **"Advanced PID control"**, *ISA-The Instrumentation, Systems, and Automation Society; Research Triangle Park*, NC 27709, 2006.

[10] Farahani, Gholamreza, **"Computing optimum coefficients of PID controller with genetic algorithm"**, I*ECON 2012-38th Annual Conference on IEEE Industrial Electronics Society. IEEE*, 2012.

[11] Saad, M. S., et al., **"Implementation of PID controller tuning using differential evolution and genetic algorithms"**, *International Journal of Innovative Computing Information and Control* Vol. 8(11), pp. 7761-7779, 2012.

[12] Fiuzy, M., Haddadnia, J., and Mousavi Mashhadi, S. K., **"Designing an Optimal PID Controller for Control the Plan's Height, Based on Control of Autopilot by using Evolutionary Algorithms",** *Journal of mathematics and computer Science*, 6 pp. 260-271, 2013.

[13] Noshahri, Hengameh, Hamed Kharrati, **"PID controller design for unmanned aerial vehicle using genetic algorithm",** *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE). IEEE*, 2014.