Journal of Engineering and Applied Sciences 14 (17): 6223-6230, 2019
ISSN: 1816-949X

Trajectory Tracking of a Tri-Copter Vertical Take-Off and Landing UAV

Nada M. Ali Ahmed and Muna H. Saleh
Department of Electrical Engineering, College of Engineering, University of Baghdad,
Baghdad, Iraq, computerengineer1986@gmail.com

Abstract: A tri-copter is a newer design of multirotor drones that can be used in narrow places and indoor environments because of its small size and weight. In this study, the model has been designed by Newton-Euler law and two controllers are used to test the model stability, the first controller is the Proportional Integral Derivative (PID) linear control and the second is the Integral Back-stepping Sliding Mode (IBSM) nonlinear control and both controller's gains have been tuned by a Particle Swarm Optimization method (PSO) to minimize the Integral Time Absolute Error (ITAE) for the system. After tuning the controllers, they are tested with step signals and many tracks have been tested with both controllers and the results are obtained in MATLAB Simulink R2017b.

Key words: Tri-copter, PID, IBSM, PSO, trajectory tracking, ITAE

INTRODUCTION

The Unmanned Aerial Vehicles (UAVs) are used for many important civilian and military applications such as area mapping, agricultural, steep terrains and rescue operations. They can be controlled either remotely or autonomously. One of the UAV's classifications is the Vertical Take-Off and Landing (VTOL) multi-rotor which uses the propellers for lifting the copter instead of swashplate like the helicopter (Wang *et al*., 2017). The VTOL multi-rotor has the skill to move between movement phases such as vertical takeoff, hover, lateral movement and landing (Sababha *et al*., 2015). There are two types of tri-copter design, the single and coaxial. The single tri-rotor which is the design depended in this study has three motors on the three arms with a servo motor on the tail arm to tilt the back motor with an accurate angle to control the yawing moment while the coaxial tri-rotor has six motors, two counter rotating motors on each arm to cancel the yaw moment on each side (Yoo *et al*., 2010).

In 1948, first three-rotor aircraft was designed and it is called "Cierva Air Horse" helicopter and it has an airshow at the Farnborough air show. It was a hugehelicopter with a possible payload of 24 travelers (Salazar-Cruz *et al*., 2008). The single tri-copter multi-rotor which is used in this researcher has less number of motors than the other multirotor types like quadcopter, hexa-copter and octacopter and this will reduce energy consumption, weight and vehicle's size (Wang *et al*., 2017). It has a triangular shape frame with four inputs which are the three motor's angular velocities and the tilt angle of the servo motor on the back arm. The system dynamical model is six degrees of freedom, high nonlinear, under actuated model such that only four control signals can be controlled directly (Kulhare *et al*., 2012).

The altering in the angular velocities of the three motors introduces the altitude, roll and pitch control actions. Increasing or decreasing all motor's speeds simultaneously will control the up-down movement. Changing in the front motor's speeds with setting the speed for the back motor will control the right-left movement. The pitch down action can be controlled by increasing the back motor's speed more than the two front motors and vice versa. And it will remain the yaw movement which changes the tri-copter head to the right or left direction and this can be produced by changing the tilt angle of the servo motor on the back arm (Yoo *et al*., 2010). Figure 1 shows the tri-copter design.

By Wang *et al*. (2017), the fuzzy backstepping sliding mode controller has been used for stabilizing the tri-rotor and then the trajectory tracking was implemented. By Kulhare *et al*. (2012) the Euler-lagrange formula has been utilized to develop the mathematical model for the tri-copter and the back-stepping based PD controller has been based to achieve the system stability at the desired hovering position. By Arega (2016), Newton-Euler formula has been used for deriving the model equations, the altitude is controlled by super twisting sliding mode control while the Euler angles are controlled with a PID control.

Corresponding Author: Nada M. Ali Ahmed, Department of Electrical Engineering, College of Engineering, University of Baghdad, Baghdad, Iraq, computerengineer1986@gmail.com

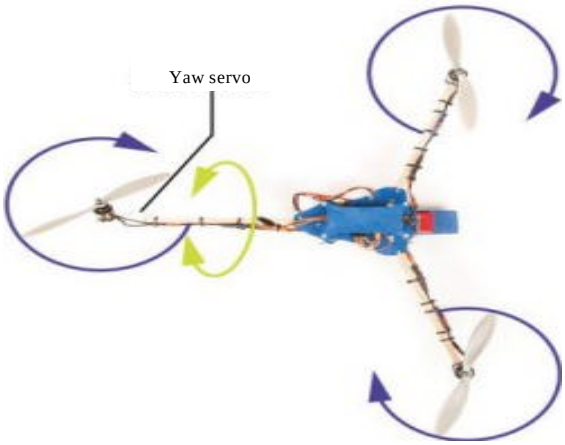

Fig. 1: Tri-copter design

MATERIALS AND METHODS

Tri-copter mathematical model: The tri-copter design is specified as a rigid body with six Degrees of Freedom (DOF) motion. In order to translate the UAV body forces to the Earth forces, the transformation matrix $R_{xyz}^{\phi\theta\psi}$ is used which considers the relation between the body coordinate system and the Earth coordinate system and as shown in Eq. 1 (Barsk, 2012):

$$R_{xyz}^{\phi\theta\psi} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi-\cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi+\cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi+\sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi-\sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix} \quad (1)$$

where, $(\phi\theta\psi)$ are Euler angles. Using Newton-Euler formula to derive the model six equations, Eq. 2 is used to derive the three translational equations and Eq. 3 is used to derive the three rotational equations (Arega, 2016):

$$m.\dot{V} = F \quad (2)$$

Where:

m = Tri-copter mass

$\dot{v}$ = The translational acceleration vector in Earth coordinate system

F = The total external Forces acting on the body

$$I.\dot{\omega}+\omega\times I\omega = \tau \quad (3)$$

Where:

I = The Inertia moments matrix

ω = The angular velocity vector in body coordinate system

τ = The total external torques acting on the tri-copter body

After completing the derivation, the full model equations are represented as follows:

$$\ddot{X} = -\frac{\sin\theta}{m}U_1 \quad (4)$$

$$\ddot{Y} = \frac{\sin\phi\cos\theta}{m}U_1 \quad (5)$$

$$\ddot{Z} = \frac{\cos\phi\cos\theta}{m}U_1 - g \quad (6)$$

$$\ddot{\phi} = \frac{I_{yy}-I_{zz}}{I_{xx}}\dot{\psi}\dot{\theta}+\frac{U_2}{I_{xx}} \quad (7)$$

$$\ddot{\theta} = \frac{I_{zz}-I_{xx}}{I_{yy}}\dot{\psi}\dot{\theta}+\frac{U_3}{I_{yy}} \quad (8)$$

$$\ddot{\psi} = \frac{I_{xx}-I_{yy}}{I_{zz}}\dot{\phi}\dot{\theta}+\frac{U_4}{I_{zz}} \quad (9)$$

Where:

I_{xx}, I_{yy}, I_{zz} = The moments of Inertia about x, y, z axes

g = The gravitational acceleration

U_1 = The altitude control

U_2 = The roll control

U_3 = The pitch control

U_4 = The yaw control

Tri-copter controllers design: The system of the tri-copter is high nonlinear with complicated dynamics, so, it needs high adaptable controllers (Wang *et al*., 2017). Figure 2 illustrates the control methodology with the system model. In this study, two controllers have been utilized separately to control the tri-copter, these are.

Proportional Integral Derivative (PID) control: The PID control is characterized by simplicity, robustness and wide usage in many applications Eq. 10 shows the general PID control equation:

$$U(t) = K_p e(t)+K_i\int_0^t e(\tau)d\tau+K_d\frac{d}{dt}e(t) \quad (10)$$

Where:

K_p, K_i, K_d = The proportional, integral, derivative gain coefficients, respectively

U(t) = The control action

e(t) = The difference between output and input signals

The K_p coefficient eliminates rise time for the system, K_i coefficient eliminates steady state error and K_d gain

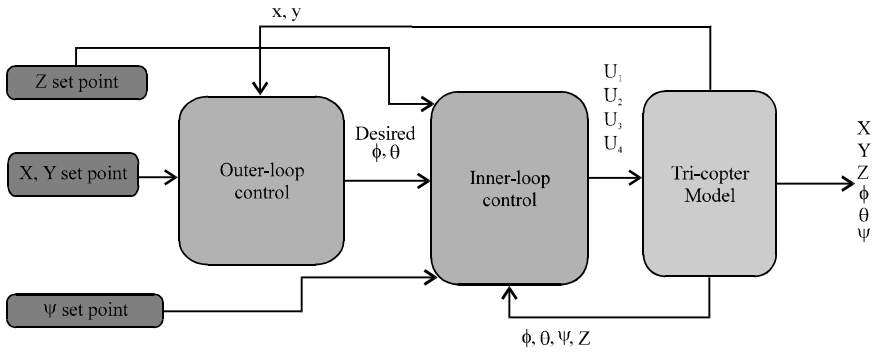

Fig. 2: Tri-copter control mechanism

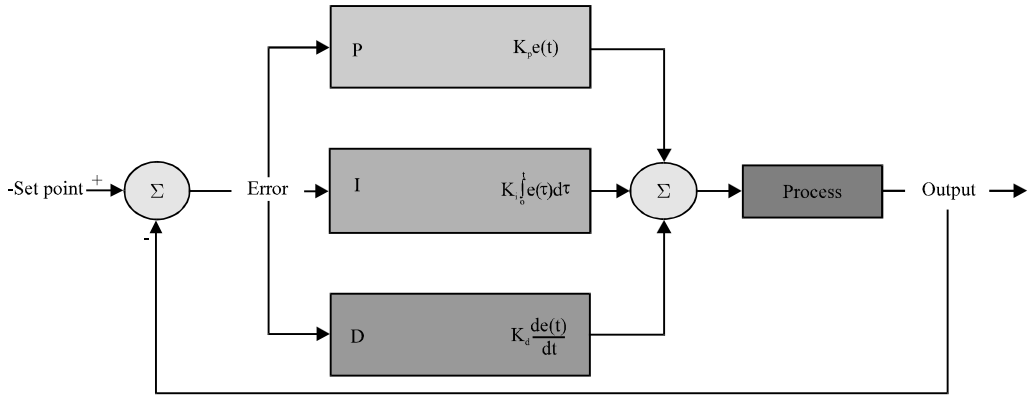

Fig. 3: PID control mechanism

eliminates the overshoot so these coefficients need to be tuned accurately to stabilize the UAV (Venkata, 2016). Figure 3 shows the control appliance.

When the PID control was added to the model, the Euler angles are assumed to be very small such that ($\sin\phi$ and $\sin\theta = 0$, $\cos\phi$ and $\cos\theta = 1$), so, the control equations are illustrated as follows:

$$U_1 = m\left(\left(K_{pz}(z_d-z)+K_{iz}\int_0^t(z_d-z)d\tau+K_{dz}\frac{d(z_d-z)}{dt}\right)+g\right) \quad (11)$$

$$U_2 = I_{xx}\left(K_{p\phi}(\phi_d-\phi)+K_{i\phi}\int_0^t(\phi_d-\phi)d\tau+K_{d\phi}\frac{d(\phi_d-\phi)}{dt}\right) \quad (12)$$

$$U_3 = I_{yy}\left(K_{p\theta}(\theta_d-\theta)+K_{i\theta}\int_0^t(\theta_d-\theta)d\tau+K_{d\theta}\frac{d(\theta_d-\theta)}{dt}\right) \quad (13)$$

$$U_4 = I_{zz}\left(K_{p\psi}(\psi_d-\psi)+K_{i\psi}\int_0^t(\psi_d-\psi)d\tau+K_{d\psi}\frac{d(\psi_d-\psi)}{dt}\right) \quad (14)$$

$$\theta_d = \frac{-m}{U_1}\left(K_{px}(x_d-x)+K_{ix}\int_0^t(x_d-x)d\tau+K_{dx}\frac{d(x_d-x)}{dt}\right) \quad (15)$$

$$\phi_d = \frac{m}{U_1}\left(K_{py}(y_d-y)+K_{iy}\int_0^t(y_d-y)d\tau+K_{dy}\frac{d(y_d-y)}{dt}\right) \quad (16)$$

where, x_d, y_d, z_d, θ_d, ψ_d are the desired signals.

Integral Back-stepping Sliding Mode (IBSM) control: One of the new nonlinear controller's designs is the Integral Back-stepping Sliding Mode (IBSM) control which is consisted from the Integral Backstepping (IB) and sliding mode controllers it can provide robustness against model uncertainties and disturbances (Bouabdallah, 2007) it also can eliminate the chattering phenomenon of the classic sliding mode control. This control can follow the tracks with minimum error. The Lyapunov stability is used in this control to prove controller stability. The system parameters are expressed as follows (Jia *et al*., 2017):

$$X = \left[x\dot{x},\ y\dot{y},\ z\dot{z},\ \phi\dot{\phi},\ \theta\dot{\theta},\ \psi\dot{\psi}\right]^T \quad (17)$$

$$X = \left[x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}\right]^T \quad (18)$$

The control method is implemented in two steps, step one is similar to the backstepping control and as follows.

Step 1: The error term and its derivative between desired and actual signals are written as (Bouchoucha *et al*., 2011):

$$z_1 = x_{1d} - x_1 \tag{19}$$

$$\dot{z}_1 = \dot{x}_{1d} - \dot{x}_1 \tag{20}$$

$$\dot{x}_1 = x_{i+1} \tag{21}$$

The first Lyapunov stability term is described as follows:

$$V_1(z_1, \xi_1) = \frac{1}{2} z_1^2 + \frac{1}{2} \lambda_1 \xi_1^2 \tag{22}$$

where, $\xi_1 = \int_0^t z_1(\tau) d\tau$, λ_1 is a positive constant, the integral term will decrease the steady state error of the system, the Lyapunov derivative will be as follows:

$$\dot{V}_1(z_1, \xi_1) = z_1 \dot{z}_1 + \lambda_1 \xi_1 \dot{\xi}_1 = z_1(\dot{z}_1 + \lambda_1 \xi_1) \tag{23}$$

The desired virtual control is given by the equation $x_{i+1} = x_{id} + \lambda_1 \xi_1 + c_1 z_1$, $c_1 > 0$, so, $\dot{V}_1 = -c_1 z_1^2$, to satisfy that $\dot{V}_1 \leq 0$ is negative semidefinite.

Step 2: In step 2, the sliding surface and its derivative will be derived and added to the controller like the following Eq. 24 and 25:

$$S_1 = z_2 = \dot{x}_{1d} + \lambda_1 \xi_1 + c_1 z_1 - x_{i+1} = \dot{z}_1 + \lambda_1 \xi_1 + c_1 z_1 \tag{24}$$

$$\dot{S}_1 = \ddot{x}_{1d} - \dot{x}_{i+1} + \lambda_1 z_1 + c_1 \dot{z}_1 \tag{25}$$

The second Lyapunov function and its time derivative are expressed as:

$$V_2(z_1, \xi_1, S_1) = \frac{1}{2} z_1^2 + \frac{1}{2} \lambda_1 \xi_1^2 + \frac{1}{2} S_1^2 \tag{26}$$

$$\dot{V}_2 = z_1 \dot{z}_1 + \lambda_1 \xi_1 \dot{\xi}_1 + S_1 \dot{S}_1 = -c_1 z_1^2 + S_1 \dot{S}_1 \text{ such that } S_1 \dot{S}_1 \leq 0 \tag{27}$$

And the sliding surface time derivative is given by the following equation which uses the saturation function instead of the signum function to eliminate chattering phenomena in the sliding surface:

$$\dot{S}_1 = -K_1 . Sat(S_1) - K_2 - S_1; K_1, K_2 > 0 \tag{28}$$

$$Sat(S_1) = \begin{cases} Sign(S_1) & 1 < S_1 < -1 \\ S_1 & -1 \leq S_1 \leq -1 \end{cases} \tag{29}$$

$$Sign(S_1) = \begin{cases} 1 & 0 < S_1 \\ 0 & S_1 = 0 \\ -1 & S_1 < 0 \end{cases} \tag{30}$$

Then, equaling Eq. 28 with Eq. 25 and substituting the x_{i+1} tearm from the model equations to conclude the control equations and as the following equations:

$$U_x = \frac{m}{U_1}\left(-\ddot{x}_{1d} - \lambda_1 z_1 - c_1 \dot{z}_1 - K_1 . Sat(S_1) - K_2 . S_1\right) \tag{31}$$

$$U_y = \frac{m}{U_1}\left(\ddot{x}_{3d} + \lambda_2 z_3 + c_2 \dot{z}_3 + g + K_3 . Sat(S_2) + K_4 . S_2\right) \tag{32}$$

$$U_1 = \frac{m}{\cos x_7 \cos x_9}\left(\ddot{x}_{5d} + \lambda_3 z_5 + c_3 \dot{z}_5 + g + K_5 . Sat(S_3) + K_6 . S_3\right) \tag{33}$$

$$U_2 = I_{xx}\left(\begin{array}{l} \ddot{x}_{7d} + \lambda_4 z_7 + c_4 \dot{z}_7 + K_7 . Sat(S_4) + \\ K_8 . S_4 - \frac{I_{yy} - I_{zz}}{I_{xx}} x_{12} x_{10} \end{array}\right) \tag{34}$$

$$U_3 = I_{yy}\left(\begin{array}{l} \ddot{x}_{9d} + \lambda_5 z_9 + c_5 \dot{z}_9 + K_9 . Sat(S_5) + \\ K_{10} . S_5 - \frac{I_{zz} - I_{xx}}{I_{yy}} x_{12} x_8 \end{array}\right) \tag{35}$$

$$U_4 = I_{zz}\left(\begin{array}{l} \ddot{x}_{11d} + \lambda_6 z_{11} + c_6 \dot{z}_{11} + K_{11} . Sat(S_6) + \\ K_{12} . S_6 - \frac{I_{xx} - I_{yy}}{I_{zz}} x_8 x_{10} \end{array}\right) \tag{36}$$

where, λ_1, c_1, $k_i > 0$ and $x_{9d} = \sin^{-1} U_x$, $X_{7d} = \sin^{-1} U_y / \cos x_9$. The tri-copter model was simulated in MATLAB Simulink, firstly with a PID control and secondly with an IBSM control separately and both controller's gains had been tuned by a PSO technique to get the best values that stabilize the UAV and eliminate the possible errors between desired and actual signals.

Particle Swarm Optimization (PSO) technique: The Particle Swarm Optimization (PSO) was developed by Kennedy and Eberhart and its operation features depend on the swarm of birds, fish or bees with no leader such that each particle located firstly with a random location then all the swarm randomly search the optimum value depending on the best local position (P_{best}), best global position (G_{best}) and their velocities. Figure 4 illustrates the PSO flowchart that is depended in programming the algorithm in an m.file in MATLAB environment to tune each controller gains separately with the below equations related to PSO algorithm (Rao, 2009).

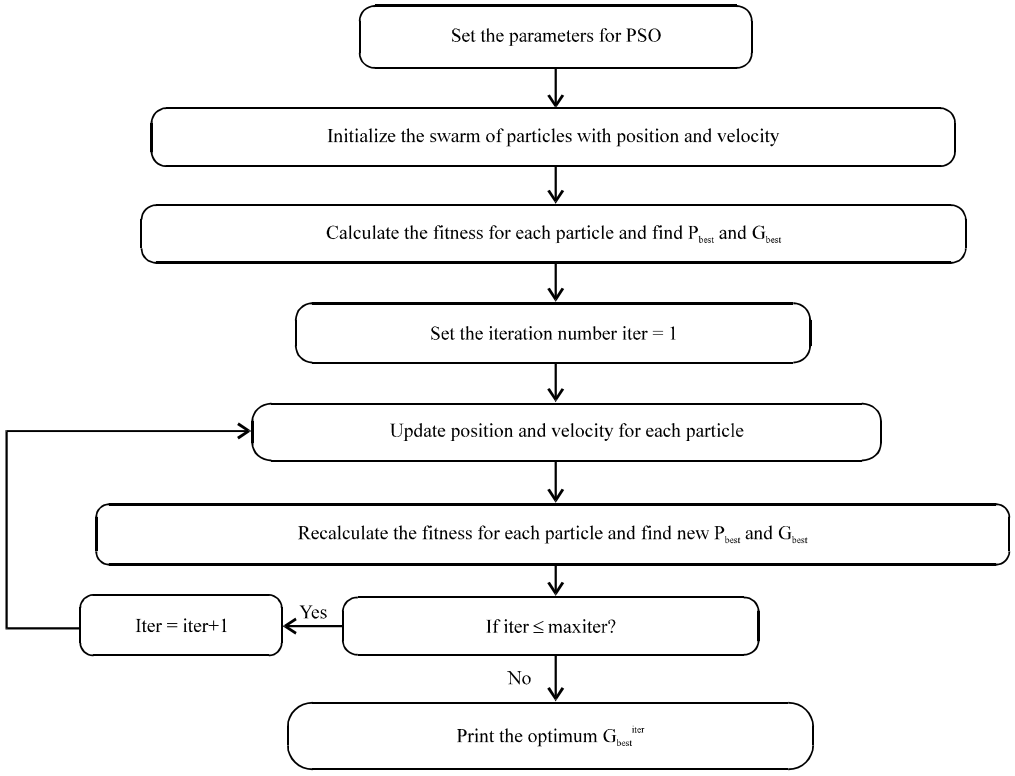

Fig. 4: Flowchart for the PSO algorithm (Alam, 2016)

$$X^{iter+1}(i,d) = X^{iter}(i,d) + V^{iter}(i,d) \quad (37)$$

$$V^{iter+1}(i,d) = \omega * V^{iter}(i,d) + C_1 * r_1\left(P_{best}^{iter}(i,d) - X^{iter}(i,d)\right) + C_2 \times r_2\left(G_{best}^{iter}(g,d) - X^{iter}(i,d)\right) \quad (38)$$

$$\omega = \omega_{max} - iter\left(\omega_{max} - \omega_{min}\right)/Maxiter \quad (39)$$

Where:

X = The position value
V = The particle Velocity is an index for the swarm size
d = An index for the number of parameters
ω = The inertia factor
$(C_1C_2r_1r_2)$ = Acceleration factors
iter = The iteration number
g = An index for the best particle with a lowest cost value
Maxiter = The maximum number of iterations

Performance index: The cost function that is depended in this researcher in the tuning procedure is the Integral Time Absolute Error (ITAE) function which gives minimum oscillation to the model with minimum overshoots and it is better than the other criterions in sensitivity and selectivity, Eq. 40 shows the cost function equation (Maiti *et al*., 2008):

$$C.F = \int_0^t \tau|e_x|d\tau + \int_0^t \tau|e_y|d\tau + \int_0^t \tau|e_z|d\tau + \int_0^t \tau|e_\phi|d\tau + \int_0^t \tau|e_\theta|d\tau + \int_0^t \tau|e_\psi|d\tau \quad (40)$$

Table 1: The tri-copter model and PSO parameters values

Symbols	Descriptions	Values/Units
m	Mass of the tri-copter	0.814 (kg)
I_{xx}	Moment of inertia on the x-axis	0.043 (kg.m^2)
I_{yy}	Moment of inertia on the y-axis	0.048 (kg.m^2)
I_{zz}	Moment of inertia on the z-axis	0.077 (kg.m^2)
I	Arm length of the tri-copter	0.15 (m)
g	Gravitational force	9.8 (m/sec^2)
C_1, C_2	Acceleration factors	2
R_1, R_2	Random variables	(0-1)
Maxiter	Maximum number of iterations	1000
ω_{max}	Maximum inertia factor	0.9
ω_{min}	Minimum inertia factor	0.4
S	Swarm size	30
d	PID parameters number	18
d	IBSM parameters number	24
t	Simulation time	10 (sec)

Table 1 shows the values of the model parameters and the optimization factors values used in this researcher.

RESULTS AND DISCUSSION

Stability results: Many iterations and runs are implemented until minimizing the ITAE values to the less value and the tuning process was stopped when there is no detected improvement in the objective function after many iterations. The tuning process is executed in two stages, firstly to the inner loop control and secondly to the outer loop control as shown in Fig. 2. Figure 5 illustrates the minimum ITAE values obtained after finishing the tuning process and testing both controllers with step inputs which indicate a sudden transition from one point to another.

Simulation results: The trajectory tracking field is the very interested topic and challenging research related with UAVs such that the UAV can be sent to do many desired missions with accurate tracking in this researcher many tracks have been tested with both used controllers in this researcher after achieving the stability for the tri-copter (Samir *et al*., 2017). Many desired tracks have been tested and simulated in Simulink with the tri-copter model with both the tuned PID and IBSM controllers to check the performance of the UAV in tracking. Any track consists of many points, each desired waypoints are given to (x_d, y_d, z_d, ψ_d) and then (ϕ_d, θ_d) are translated from (x, y) control equations. The desired and actual trajectory tracks are illustrated as follows.

Point to point track: This track is useful for sending the UAV to deliver some payloads from one point to another. The tri-copter takes off to 5 m height and it is moving in a lateral path from (x, y) = 0, 0-5, 5 and then it lands down. Figure 6 illustrates the 3D and 2D simulation point to point track for the tri-copter with the waypoints defined in Table 2.

Square track: This track can be used for detecting a specific region in a square path. The tri-copter takes off to 5 m height and follows the path of square shape with 5 m for each side and then return to the first point and lands down. Figure 7 illustrates the 3D and 2D square path with the waypoints specified in Table 3.

Table 2: Point to point track way points

Start point (x, y, z)	End point (x, y, z)
(0, 0, 0)	(0, 0, 5)
(0, 0, 5)	(5, 5, 5)
(5, 5, 5)	(5, 5, 0)

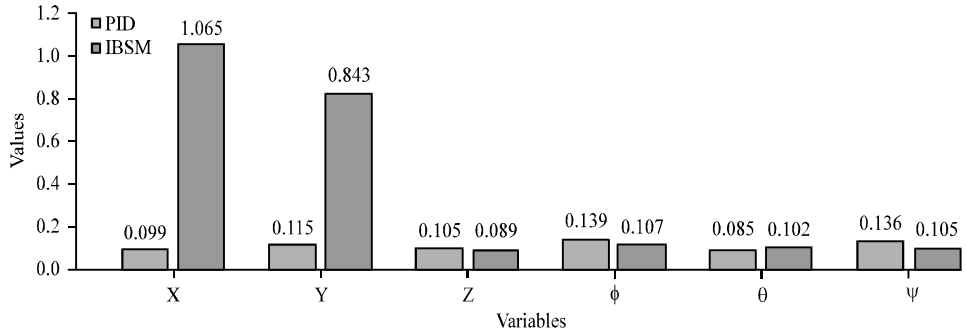

Fig. 5: ITAE values for both controllers

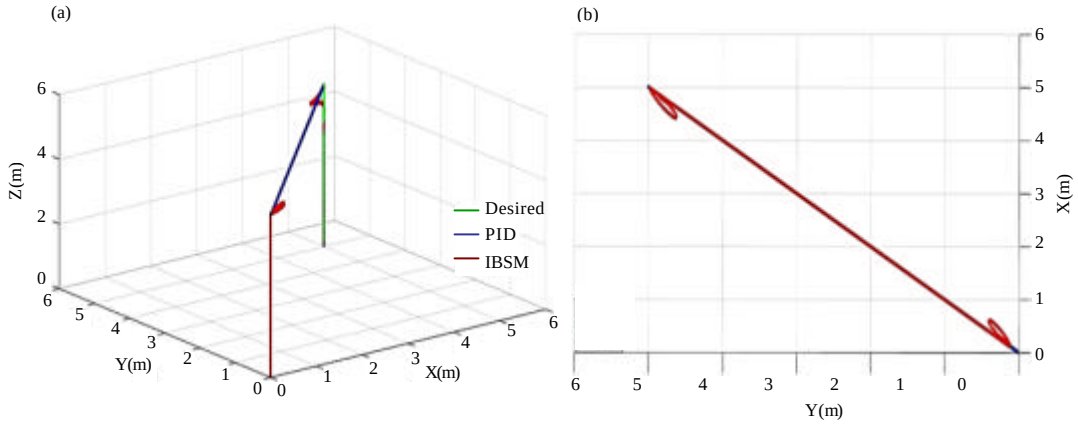

Fig. 6: a, b) 3D and 2D point to point track depending on PID, IBSM controllers

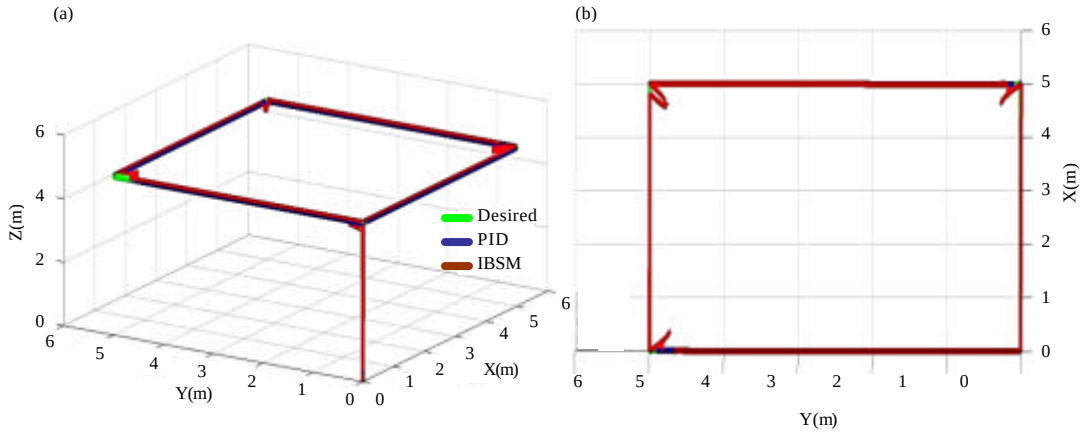

Fig. 7: a, b) 3D and 2D square track depending on PID, IBSM controllers

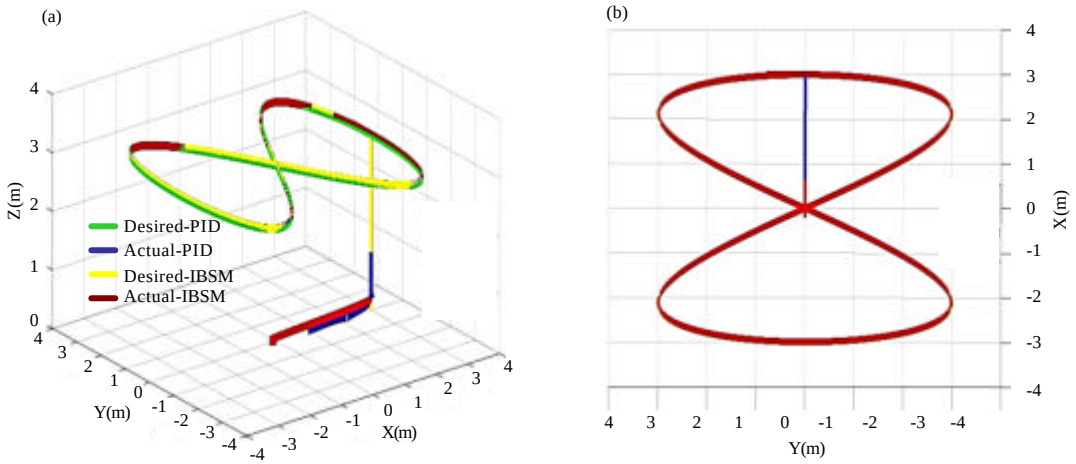

Fig. 8: a, b) 3D and 2D eight figure track depending on PID, IBSM controllers

Table 3: Square track way points

Start point (x, y, z)	End point (x, y, z)
(0, 0, 0)	(0, 0, 5)
(0, 0, 5)	(5, 0, 5)
(5, 0, 5)	(5, 5, 5)
(5, 5, 5)	(0, 5, 5)
(0, 5, 5)	(0, 0, 5)
(0, 0, 5)	(0, 0, 0)

Eight figure track: In this track, the tri-copter takes off to 3 m height and follows the desired equations as follows, Fig. 8 shows the track waypoints in 3D and 2D dimensions:

$$\left.\begin{aligned} X_d(t) &= 3\cos(0.025\pi)t \\ Y_d(t) &= 3\sin(0.05\pi)t \\ Z_d(t) &= 3 \\ \psi_d(t) &= 0 \end{aligned}\right\} \text{for PID control} \qquad (41)$$

$$\left.\begin{aligned} X_d(t) &= 3\cos(0.004\pi)t \\ Y_d(t) &= 3\sin(0.008\pi)t \\ Z_d(t) &= 3 \\ \psi_d(t) &= 0 \end{aligned}\right\} \text{for IBSM control} \qquad (42)$$

CONCLUSION

In this study, a tri-copter UAV has been tested with two types of controllers PID and IBSM and both showed minimum error ITAE values for the stability results when step signals have been applied to the system. As the system now is stable, tri-copter can be sent to do multiple different tracks by applying desired waypoints. Many different tracks are tested in this researcher and both PID and IBSM controllers show good tracking results with minimum error but with different consumed times needed to complete the desired tracks.

REFERENCES

Alam, M.N., 2016. Particle swarm optimization: Algorithm and its codes in MATLAB. J. Anal. Sci., 1: 1-10.

Arega, A.B., 2016. Addis Ababa Institute of Technology Department of Electrical and Computer Engineering. Ph.D Thesis, Addis Ababa University, Addis Ababa, Ethiopia.

Barsk, K.J., 2012. Model predictive control of a tricopter. MSc Thesis, Linkopings University, Linkoping, Sweden.

Bouabdallah, S., 2007. Design and control of quadrotors with application to autonomous flying. Ph.D Thesis, EPFL, Switzerland.

Bouchoucha, M., S. Seghour, H. Osmani and M. Bouri, 2011. Integral backstepping for attitude tracking of a quadrotor system. Electron. Electr. Eng., 116: 75-80.

Jia, Z., J. Yu, Y. Mei, Y. Chen and Y. Shen *et al*., 2017. Integral backstepping sliding mode control for quadrotor helicopter under external uncertain disturbances. Aerosp. Sci. Technol., 68: 299-307.

Kulhare, A., A.B. Chowdhury and G. Raina, 2012. A back-stepping control strategy for the Tri-rotor UAV. Proceedings of the 24th Chinese Control and Decision Conference (CCDC), May 23-25, 2012, IEEE, Taiyuan, China, ISBN:978-1-4577-2073-4, pp: 3481-3486.

Maiti, D., A. Acharya, M. Chakraborty, A. Konar and R. Janarthanan, 2008. Tuning PID and FOPID controllers using the integral time absolute error criterion. J. Comput. Sci., 1: 1-6.

Rao, S.S., 2009. Engineering Optimization: Theory and Practice. Wiley, New York.

Sababha, B.H., H.M.A. Zubi and O.A. Rawashdeh, 2015. A rotor-tilt-free tricopter UAV: Design, modelling and stability control. Intl. J. Mechatron. Autom., 5: 107-113.

Salazar-Cruz, S., F. Kendoul, R. Lozano and I. Fantoni, 2008. Real-time stabilization of a small three-rotor aircraft. IEEE. Trans. Aerosp. Electron. Syst., 44: 783-794.

Samir, A., A. Hammad, A. Hafez and H. Mansour, 2017. Quadcopter trajectory tracking control using state-feedback control with integral action. Intl. J. Comput. Appl., 168: 1-7.

Venkata, V.P., 2016. Control system design for tricopter using filters and PID controller. J. Eng., 1: 1-5.

Wang, S., J. Zhang, Q. Zhang and C. Pei, 2017. An innovative fuzzy backstepping sliding mode controller for a Tri-Rotor Unmanned Aerial Vehicle. Microsyst. Technol., 23: 5621-5630.

Yoo, D.W., H.D. Oh, D.Y. Won and M.J. Tahk, 2010. Dynamic modeling and stabilization techniques for tri-rotor unmanned aerial vehicles. Intl. J. Aeronaut. Space Sci., 11: 167-174.