Self Tuning Fuzzy PD Control for Quadcopter

Muhammad Bilal Kadri[1], Faizan Shahid[2]

[1] Mechatronics Department, College of Engineering, PAF-KIET, Karachi 75190, Pakistan

[2] PNEC, EPE Department, National University of Science & Technology, Karachi 75350,

Received: 01 February, 2019 / Revised: 20 April, 2019 / Accepted: 14 June, 2019

ABSTRACT

The prime objective of this paper is to study the effectiveness of self-tuning fuzzy PD (STFPD) control scheme for a quadcopter model having six degrees of freedom (DoF). The control performance of STFPD is compared with conventional proportional derivative (PD) controller. Quadcopter model considered in this work is a nonlinear multi input multi output (MIMO) system with coupled dynamics. In STFPD two layered architecture is implemented. The low-level control is employed to stabilize the quadcopter and is termed as the attitude control whereas the higher-level control ensures trajectory tracking. Feedback sensor noise is introduced to make simulation more practical and realistic. Controllers are implemented in Matlab/Simulink platform. Simulation results demonstrates the efficacy of STFPD over the conventional control scheme.

KEYWORDS

Quadcopter Model; Autonomous Attitude Control; Self tuning fuzzy PD control

I. Introduction

Quadcopters have myriad civil and military applications including but not limited to rescue, monitoring and surveillance, they have been studied extensively. Quadcopters are a class of unmanned air vehicles (UAV) which consist of four rotors which enable it for vertical take-off and landing (VTOL). Technology has evolved at an accelerated pace and advances in the field of unmanned control systems have seen exponential growth. Many national and international R&D organizations are currently working to develop indigenous unmanned control system integrated with mobile robots. Due to the complexity of the quadcopter dynamics many models have been proposed [1-4]. A plethora of control strategies ranging from PID controllers to more advanced non-linear control schemes such as H∞, back-stepping and self-tuning fuzzy logic controllers have been reported in literature [3-6]. Proportional derivative (PD) controller has been applied for stabilization of the quadcopter by [4-5,9-10]. In Ref [4], after initial stabilization of the quadcopter dynamics a heuristic scheme is applied for trajectory tracking, the effect of random fluctuations is rejected by employing another PD with the heuristic scheme. In [9] a fuzzy self-tuning PD controller is proposed for quadcopter stabilization and trajectory tracking. Since the quadcopter dynamics varies with the operating condition three independent PD controllers are designed for roll, pitch and yaw. In this work comparison between PD and self-tuning fuzzy PD (STFPD) controllers is investigated. PD controllers for attitude and trajectory are designed whereas in STFPD the fuzzy system update the gains of the outer loop control i.e. trajectory tracking controller.

Quadcopter Model

Quadcopter consists of four rotors assembled in cross configuration having six degrees of freedom (DoF).The system is under actuated, hence difficult to control. Roll, pitch and yaw movement can be achieved by different configuration of rotor speeds and directions. Complete details of the quadcopter maneuverability can be found in [2-4]. The thrust generated by the four rotors (as shown in Fig 1) is represented by $\tau_L, \tau_R, \tau_F, \tau_B$ respectively. The angular speed which generates angular momentum at the center of rotation is given by $\Omega_L, \Omega_R, \Omega_F, \Omega_B$ respectively

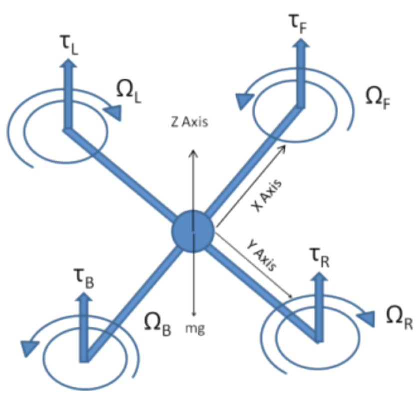

Fig 1: Quadcopter structure

Dynamics of quadcopter are derived from the absolute angles ($\emptyset$, θ, φ) in body fixed frame.

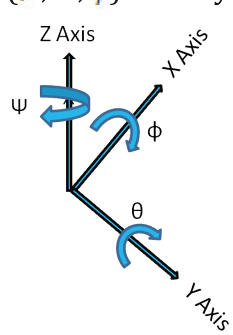

Fig 2: Quadcopter degrees of motion

State variables for quadcopter's dynamic model are as follows

x = Position at x-axis

y = Position at y-axis

z = Position at z-axis

$\dot{x}$ = Velocity in x-axis direction

$\dot{y}$ = Velocity in y-axis direction

$\dot{z}$ = Velocity in z-axis direction

$\ddot{x}$ = Linear Acceleration in x direction

$\ddot{y}$ = Linear Acceleration in y direction

$\ddot{z}$ = Linear Acceleration in z direction

$\emptyset$ = Absolute roll angle

θ = Absolute pitch angle

φ = Absolute yaw angle

$\dot{\emptyset}$ = Roll rate

$\dot{\theta}$ =Pitch Rate

$\dot{\varphi}$ =Yaw Rate

p = Roll rate in body fixed frame

q = Pitch rate in body fixed frame

r = Yaw rate in body fixed frame

Rotor dynamics are kept simple, BLDC with its propeller is modelled as a first order transfer function with a time constant of 0.1 sec.

$$G(s) = \frac{1}{0.1s + 1} \quad (1)$$

Dynamic equations of quadcopter are as follows [9]

$$\begin{bmatrix} \dot{\emptyset} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\emptyset \tan\theta & \cos\emptyset \tan\theta \\ 0 & \cos\emptyset & -\sin\emptyset \\ 0 & \frac{\sin\emptyset}{\cos\theta} & \frac{\cos\emptyset}{\cos\theta} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix}_v = -\frac{T}{m} \begin{bmatrix} \cos\emptyset \sin\theta \cos\varphi + \sin\theta \sin\varphi \\ \cos\emptyset \sin\theta \sin\varphi + \sin\emptyset \cos\varphi \\ \cos\emptyset \cos\theta \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \left(\frac{J_y - J_z}{J_x}\right) qr \\ \left(\frac{J_z - J_x}{y}\right) pr \\ \left(\frac{J_x - J_y}{J_z}\right) pq \end{bmatrix} + \begin{bmatrix} J_r \theta^{\cdot} \Omega_r \\ -J_r \theta^{\cdot} \Omega_r \\ 0 \end{bmatrix} + \begin{bmatrix} \frac{\tau_\varphi}{J_x} \\ \frac{\tau_\theta}{J_y} \\ \frac{\tau_\varphi}{J_z} \end{bmatrix} \quad (4)$$

Where T and $\tau_\emptyset, \tau_\theta, \tau_\varphi$ are required thrust and torques to achieve the desired trajectory. Eq. (2) is an inverse of rotational matrix from inertial/ vehicle frame to body fixed frame showing the relationship between absolute angles (Ø, Ɵ, φ) and angular rates (p, q, r). Eq. (3) is simplified form of aerodynamics/ linear acceleration of quadcopter obtained by employing a translational and rotational matrix. In Eq. (4) gyroscopic effects are given $(J_y - J_z)qr$ as roll angle, $(J_z - J_x)pr$ pitch angle and $(J_x - J_y)pq$ as yaw angle effect. Where $J_x, J_y \& J_z$ represents rotational inertia of quadcopter along x, y and z axes respectively.

$$\Omega_r = \Omega_R - \Omega_F + \Omega_L - \Omega_B \quad (5)$$

Where Ω representing rotor speed and Ωris overall residual propeller angular speed.

Controller Design

Control design of quadcopter involves developing a flight control algorithm for under-actuated four rotors responsible for controlling Quadcopter's 6 DoF i.e. its position and orientation. The responsibility for flight control system is to maintain orientation (Ø, Ɵ, φ), while moving towards the desired location (xd, yd, zd). Simple PD Control strategy has been developed as shown in figure 3 & 4. Inner loop stabilizes the orientation of craft and outer loop guarantees the quadcopter to achieve the desired trajectory.

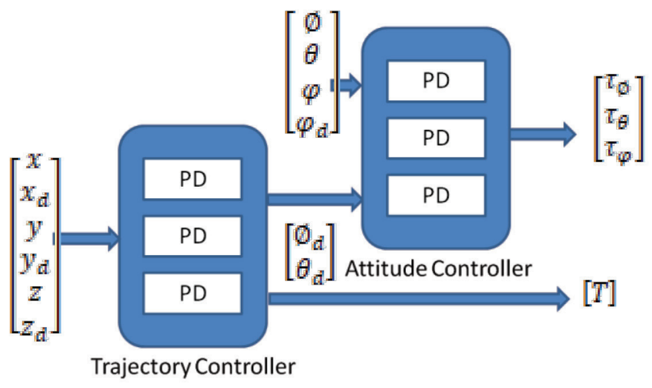

Fig 3: Controller flow chart

The System can be rewritten as $\dot{X} = f(U, X)$. The state vector involved in control design is as follows.

$$X = [\varnothing \;\; \dot{\varnothing} \;\; \theta \;\; \dot{\theta} \;\; \varphi \;\; \dot{\varphi} \;\; x \;\; \dot{x} \;\; y \;\; \dot{y} \;\; z \;\; \dot{z}] \quad (6)$$

III.A. PD controller:

Proportional Derivative control is one of the most simplistic and widely implemented control techniques. This control technique requires full state feedback from the sensors. Using state feedback, the controller calculates the difference between desired and current output and adjusts controlled input (U(t)) accordingly. The equation of controller is as follows [3].

$$U(t) = Kp\, e(t) + Kd\, \frac{de(t)}{dt} \quad (6)$$

III.B Trajectory Controller in outer loop:

Trajectory controller is a set of three PD controllers for which mathematical equations are as follows.

$$e_x = x - x_d$$

$$\dot{e}_x = \dot{x} - \dot{x}_d$$

Similarly ey and ez can be calculated. The control law is given by:

$$U_x = Kp_x\, e_x + Kd_x\, \dot{e}_x \quad (7)$$

Similarly for Y-axis and Z-axis motion

$$U_y = Kp_y\, e_y + Kd_y\, \dot{e}_y \quad (8)$$

$$U_z = Kp_z\, e_z + Kd_z\, \dot{e}_z \quad (9)$$

The flight control algorithm in outer loop proceeded by equations directly derived from dynamics equations. Simplified forms of Eq. 1 to 3 are as follows [9].

$$\ddot{x} \triangleq -\frac{T}{m}\cos\varnothing\sin\theta \quad (10)$$

$$\ddot{y} \triangleq \frac{T}{m}\sin\varnothing \quad (11)$$

$$\ddot{z} \triangleq g - \frac{T}{m}\cos\varnothing\cos\theta \quad (12)$$

$$\begin{bmatrix}\ddot{\varnothing}\\ \ddot{\theta}\\ \ddot{\varphi}\end{bmatrix} = \begin{bmatrix}\frac{\tau_\varphi}{J_x}\\ \frac{\tau_\theta}{J_y}\\ \frac{\tau_\varphi}{J_z}\end{bmatrix} \quad (13)$$

To obtain
$\ddot{x} = U_x, \ddot{y} = U_y, \ddot{z} = U_z$

Desired roll angle ($\emptyset_d$) and pitch angle (θ_d) can be

computed, substituting $\ddot{z} = U_z$ in Eq. 8 for $\frac{T}{m}$.

$$\frac{T}{m} = \frac{g - U_z}{\cos\emptyset \cos\theta} \quad (14)$$

Solving Eq. 6 & 7 for and respectively by substituting Eq. (10).

$$\emptyset_d = \tan^{-1}\left(\frac{U_y \cos\theta_d}{g - U_z}\right) \quad (15)$$

Where is as follows

$$\theta_d = \tan^{-1}\left(\frac{U_x}{U_z - g}\right) \quad (16)$$

III.C Attitude Controller in inner loop:
Attitude controller is also a PD controller for which mathematical equations are as follows.

$$e_\emptyset = \emptyset - \emptyset_d$$

$$\dot{e_\emptyset} = \dot{\emptyset} - \dot{\emptyset_d}$$

$$\tau_\emptyset = \frac{I_{xx}}{L}(Kp_\emptyset\, e_\emptyset + Kd_\emptyset\, \dot{e}_\emptyset) \quad (17)$$

Similarly, for pitch and yaw angle

$$\tau_\theta = \frac{I_{yy}}{L}(Kp_\theta\, e_\theta + Kd_\theta\, \dot{e}_\theta) \quad (18)$$

$$\tau_\varphi = \frac{I_{zz}}{L}(Kp_\varphi\, e_\varphi + Kd_\varphi\, \dot{e}_\varphi) \quad (19)$$

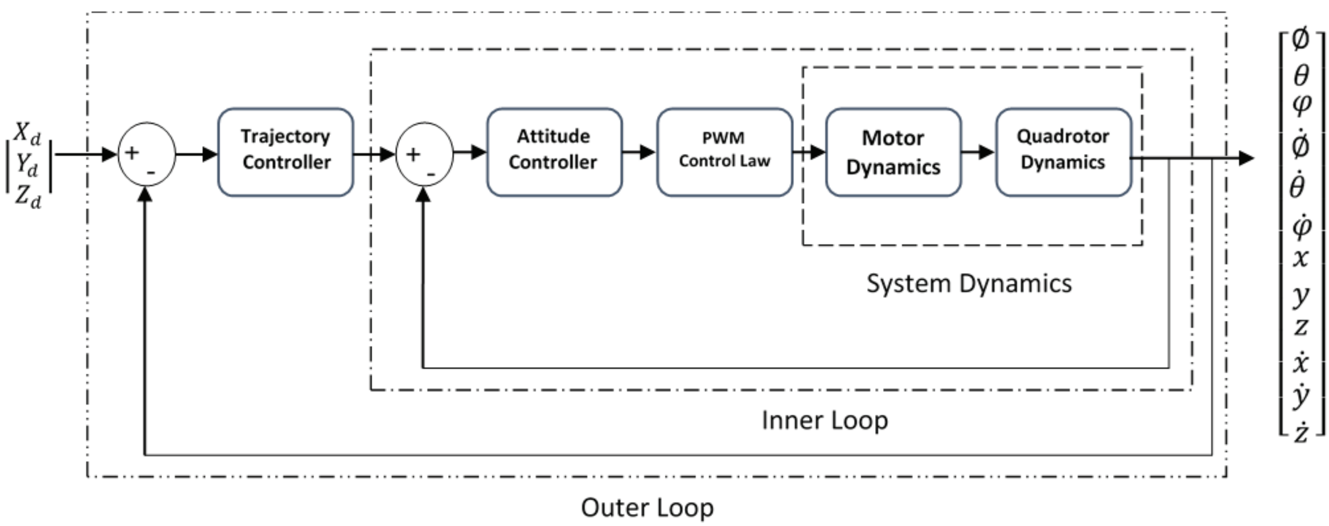

Fig 4 System block diagram using PD scheme

III.D. PWM Control Law:

Motors of Quadcopter take input as PWM whereas the controller output is in the form of required thrust and torques to stabilize the orientation of the craft and achieve desired/ target position. PWM control law is a translation between thrust and torques to the PWM pulses for rotors. It can be described by the following mathematical equations [4].

$$T_{mot} = K_T\, PWM_{mot} \quad (20)$$

$$\tau_{mot} = K_\tau\, PWM_{mot} \quad (21)$$

Where K_T and K_τ are motor specific constants.

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = G \times \begin{bmatrix} T \\ \tau_\emptyset \\ \tau_\theta \\ \tau_\varphi \end{bmatrix} \quad (22)$$

With

$$G = \begin{bmatrix} K_T & K_T & K_T & K_T \\ 0 & -l \times K_T & 0 & l \times K_T \\ l \times K_T & 0 & -l \times K_T & 0 \\ K_\tau & K_\tau & K_\tau & K_\tau \end{bmatrix}^{-1} \quad (23)$$

Where is the length of arm/ wing.

Motors have saturation in terms of speed. It is important to apply saturation nonlinearity to make the system more realistic.

$$\begin{bmatrix} PWM_F \\ PWM_R \\ PWM_B \\ PWM_L \end{bmatrix} = Sat \begin{bmatrix} U_1 + offset \\ U_2 + offset \\ U_3 + offset \\ U_4 + offset \end{bmatrix} \quad (24)$$

Where offset is defined a priori, this bias is introduced to counter balance the weight of Quadcopter and resultant PWM pulses are saturated to the maximum threshold of speed of the motors.

Self-Tuning Fuzzy PD (Stfpd) Control

In Self-Tuning Fuzzy PD (STFPD) control design Mamdani fuzzy controller is employed as a supervisory controller where it is applied to calculate control gain for each PD controller used to achieve desired position. Three fuzzy controller each having two inputs namely error and rate of change of error and two outputs named Proportional gain and Derivative gain (K_p, K_d) are given below.

$$(K_{px}, K_{dx}) = FLC_x(e_x, \dot{e}_x) \quad (25)$$

$$(K_{py}, K_{dy}) = FLC_y(e_y, \dot{e}_y) \quad (26)$$

$$(K_{pz}, K_{dz}) = FLC_z(e_z, \dot{e}_z) \quad (27)$$

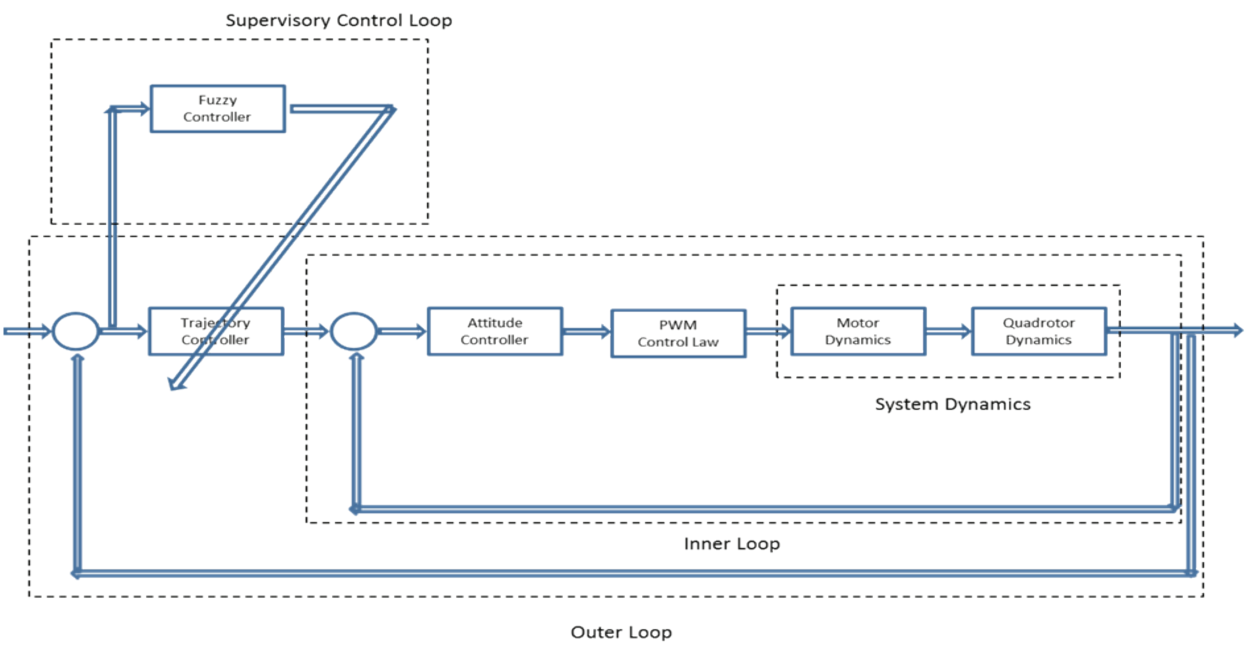

Fig 5: System block diagram using STFPD scheme

Input membership functions are same for error and rate of change of error and comprises of trapezoidal and triangular membership function. All the inputs have been normalized in the range of -1 to 1.

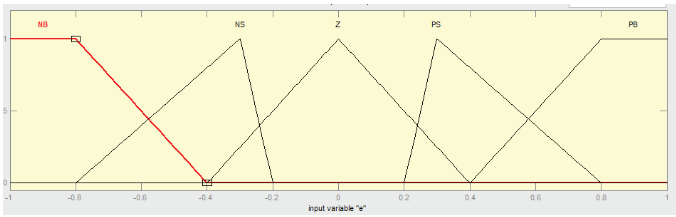

Figure 6: Input membership functions for error & rate of change of error (ė)

Unified rule base designed for consisting of 25 IF-THEN rules using prior knowledge are presented below. The rule base is not symmetric.

e	NB	NS	Z	PS	PB
NB	PVB	PVB	PVB	PB	PM
NS	PVB	PVB	PB	PB	PM
Z	PB	PB	Z	PS	PS
PS	PM	PS	PS	PS	PS
PB	PS	PS	Z	Z	Z

Table I: Rule Base For K_P

e \ ė	NB	NS	Z	PS	PB
NB	Z	Z	PS	PS	PB
NS	Z	Z	Z	Z	PS
Z	Z	Z	Z	PS	PB
PS	PS	PS	PS	PB	Z
PB	Z	Z	Z	PB	PB

Table II: Rule base for k_d

Output membership functions are same for Proportional and Derivative gains and consist of trapezoidal and triangular functions. Range of output variable is 0 to 1.

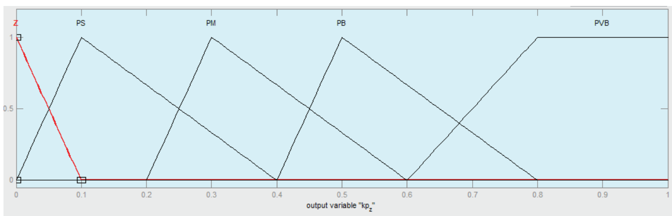

Figure 7: output membership functions for error & rate of change of error (ė)

Centre of Gravity (CoG) defuzzification method is used to convert fuzzy output membership value into a crisp value.

$$y_q^{crisp} = \sum_{i=1}^{R} b_i^q \int \mu_{B_q^i}(y_q)\, dy_q \qquad (28)$$

Simulation

Simulation of attitude and trajectory control of Quadcopter shown in fig 8, has been performed in Matlab/ Simulink platform. The output of system dynamics block is accelerations (linear and angular) and velocity (linear and angular) that are further integrated to obtain velocity (linear and angular) and position (linear and angular). Initial conditions for all state variables are set to zero.

Sensor noise is induced in the system to make simulation more realistic. Simulation has been performed using conventional PD and STFPD control schemes for way point trajectory tracking. With conventional PD control K_p & K_d gain values are fixed while in STFPD scheme K_p & K_d gains changes according to the varying operating conditions. The gains are adjusted in order to achieve a smooth transition between different phases of operation. Gains of the conventional PD controller used in the simulation are given in Table III. Universal and Quadcopter structural constants used in simulation are given in Table IV.

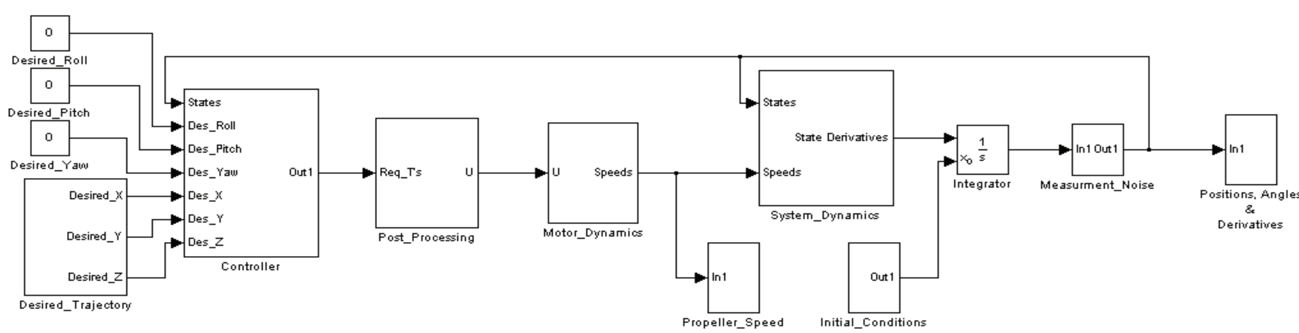

Figure 8: Simulation Model for Trajectory Tracking Control of Quadcopter

Variable	Parameter Value	
i	$K_{i,p}$	$K_{i,d}$
x	1.18	4
y	1.1	1.5
z	1	1
Ø	3	1.5
θ	1.5	2
φ	1.2	2

Table III: PD controller parameter values

S.no	Parameter	Value	Unit
1	Gravitational acceleration (g)	9.8	m/s²
2	Mass of Quadcopter (m)	0.4794	Kg
3	Length of wings (l)	0.225	m
4	Rotational Inertia along x-axis (J_x)	0.0086	Kg.m²
5	Rotational Inertia along y-axis (J_y)	0.0086	Kg.m²
6	Rotational Inertia along z-axis (J_z)	0.0172	Kg.m²
7	Residual Rotational Inertia(J_r)	3.7404×10^{-5}	Kg.m²
8	*offset*	0	rad/s
9	Motor constant (K_T)	3.13×10^{-5}	-
10	Motor constant (K_τ)	9×10^{-7}	-

Table IV: Quadcopter's dynamic constant parameter for simulation

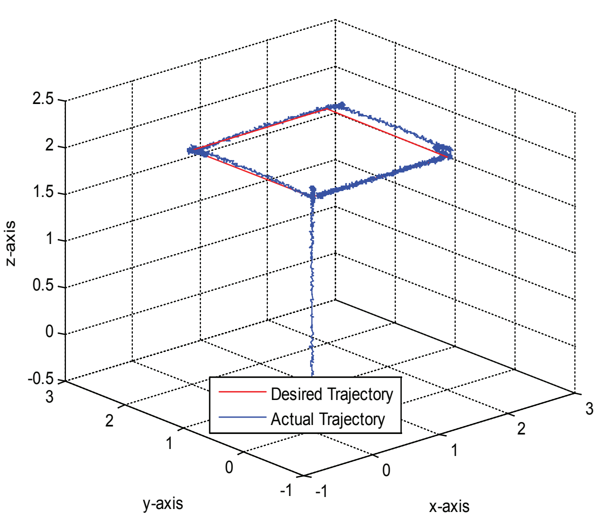

Figure 9: UAV trajectory results using PD controller for way point trajectory

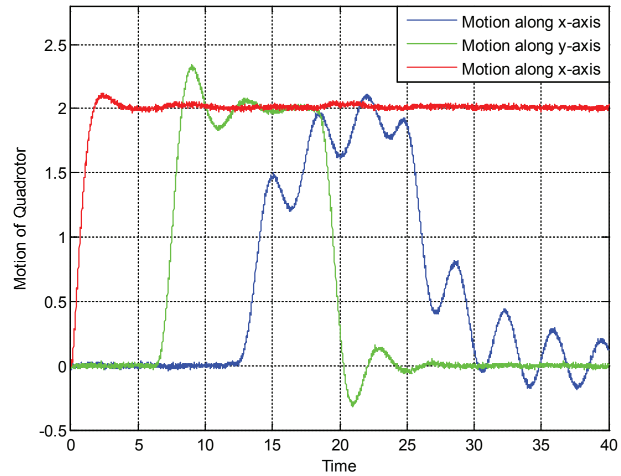

Figure 10: UAV motion along x, y and z axes using PD controller for way point trajectory

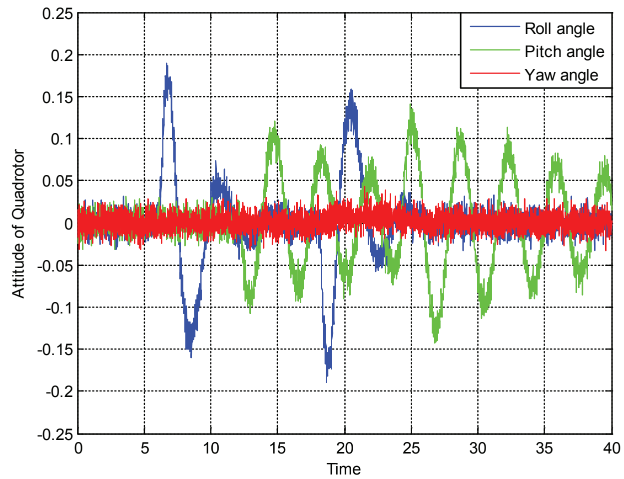

Figure 11: UAV attitude angles using PD controller for way point trajectory

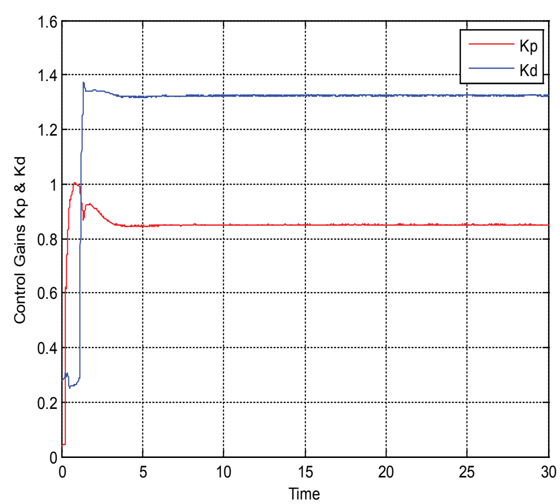

Figure 12: Adaptation of control gains by STFPD control scheme

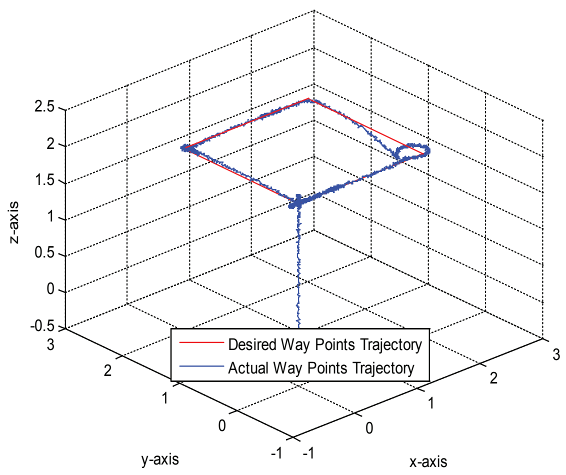

Figure 13: UAV trajectory results using STFPD controller for way point trajectory

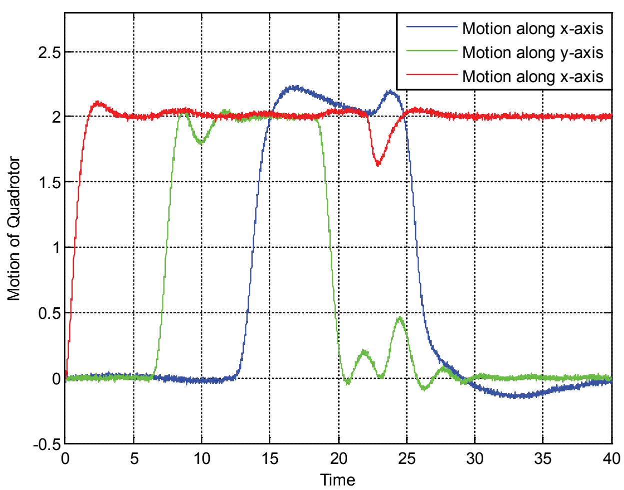

Figure 14: UAV motion along x, y and z axes using STFPD controller for way point trajectory

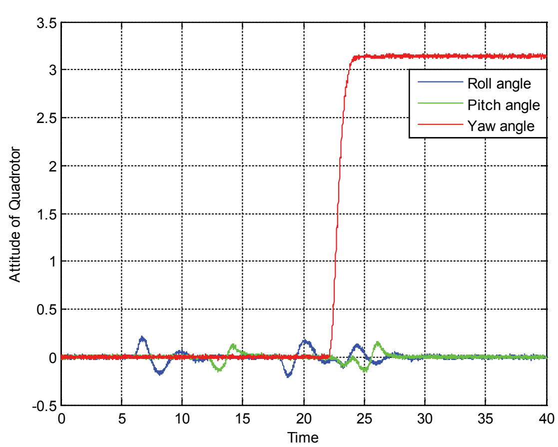

Figure 15: UAV attitude angles using PD controller for way point trajectory

Trajectory tracking with the conventional PD and STFPD are shown in Fig 9 and Fig 13 respectively. With both the controllers the quadcopter is able to achieve the desired trajectory. It can be observed from figure 13 that there are initial transient error which subsides with time. Trajectory tracking with STFPD is much more robust as compared to the conventional PD-controller. Variation in the gains of the STFPD controller with time are show in figure 12. The attitude i.e. roll, pitch and yaw angles of the quadcopter with the PD and STFPD controller are shown in figure 11 and 15 respectively. With conventional PD controller the roll and pitch oscillates throughout the motion trajectory whereas with the STFPD the roll and pitch are relatively smooth.

RMSE Based Comparison

RMSE based comparison of the results discussed above are presented in this section. As can be seen from Figure 16 that the quadcopter with STFPD controller was able to achieve the desired reference trajectory in the x-direction with a slight overshoot. On the contrary PD controller oscillates throughout the upward and downward motion. When the x-direction of the quadcopter changes, STFPD outperforms the conventional PD controller. Motion in the y and z

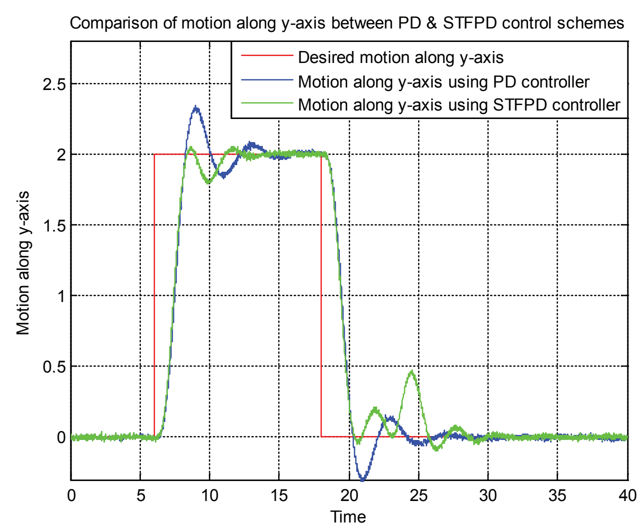

Figure 17: Comparison of motion along y-axis between PD & STFPD control schemes

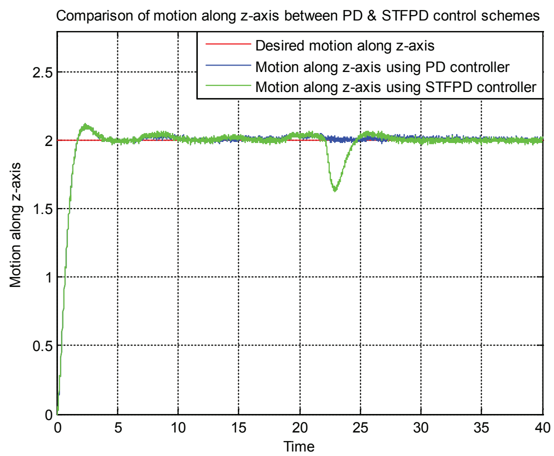

Figure 18: Comparison of motion along z-axis between PD & STFPD control schemes

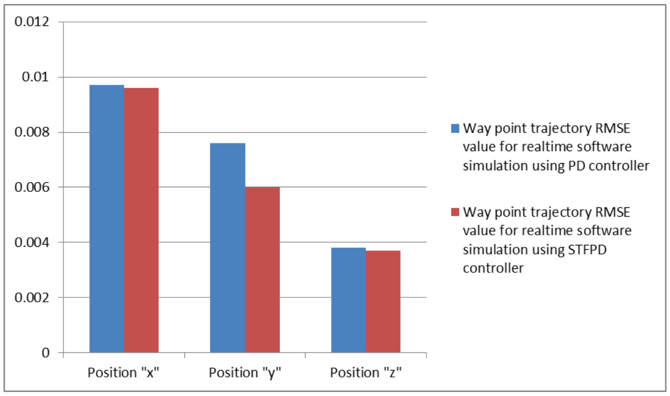

Figure 19 RMSE comparison chart between PD and STFPD control schemes

Conclusion

In this paper autonomous attitude and trajectory control problem is being addressed. Quadcopter model used in simulation have 6 DoF with coupled dynamics. The motion of the machine is primarily the function of attitude angles (Ø, Ɵ, φ), PD controller is employed to achieve the target position as well as to maintain orientation, oscillations can be observed in the simulation results. But when STFPD controller is employed UAV model produced good flight trajectories as well as oscillation has been removed. The performance of overall system can be optimized using more advance control schemes such as MPC, Robust Multivariable and FMRLC. It is quite evident that the designed STFPD scheme has superior performance than the simple PD control. RMSE values for STFPD are lesser than PD scheme. STFPD scheme has also removed oscillation from the system on the other hand response is quick as well.

References

[1] Bouabdallah , S.,Noth, A., Siegwart, R., “PID vs LQ control techniques applied to an indoor micro quadcopter,” IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3, pp. 2451–2456, 2004.

[2] Hoffmann, G.M., Huang, H. , Waslander, S.L., Tomlin, C.J., “Quadcopter helicopter flight dynamics and control: Theory and experiment,” Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, 2007.

[3] Basta,P.O.,“Quadcopter Flight”MS Thesis, California State University, Northridge, 2012

[4] Luukkonen, T., “Modelling and control of quadcopter”, School of Science, Mat-2.4108, Independent research project in applied mathematics,Espoo, August 22, 2011

[5] Erginer, B., Altuğ, E., “Modeling and PD Control of a Quadcopter VTOL Vehicle” Proceedings of the 2007 IEEE Intelligent Vehicles Symposium Istanbul, Turkey, June , 2007.

[6] Kirli, A., Ömürlü, V.E., Büyüksahin, U., Artar, R., Ortak,E., "Self tuning fuzzy PD application on TI TMS320F28335 for an experimental stationary quadcopter" 4th European Education and Research Conference (EDERC 2010), 2010,Pages: 42 - 46

[7] Zhang, Y.M., Chamseddine, A., Rabbath, C.A., Gordon, B.W., Su, C.Y., Rakheja,S., Fulford, C., Apkarian, J., Gosselin, P., “Development of advanced FDD and FTC, techniques with application to an unmanned quadcopter helicopter testbed”, Journal of the Franklin Institute, Volume 350, Issue 9, November 2013, Pages 2396-2422

[8] Ponce, P., Molina, A., Cayetano, I., Gallardo, J., Rodriguez, J.,“Experimental Fuzzy Logic Controller Type 2 for a Quadcopter Optimized by ANFIS”IFAC, Volume 48, Issue 3, 2015, Pages 2435-2441

[9] Yanjun, L., Tianqi, X.; Xiaodong, Z., "A fuzzy self-tuning PD controller for a quadcopter: Design and implementation" 2016 Chinese Control and Decision Conference (CCDC), 2016, Pages: 2448 - 2453

[10] Randal W. Beard Brigham Young University, “Quadcopter Dynamics and Control”, October 3, 2008