 CMC, vol.57, no.3, pp.365-388, 2018

A Bio-Inspired Global Finite Time Tracking Control of Four-Rotor Test Bench System

Rooh ul Amin[1], Irum Inayat[2], Li Aijun[1], Shahaboddin Shamshirband[3, 4, *] and Timon Rabczuk[5]

Abstract: A bio-inspired global finite time control using global fast-terminal sliding mode controller and radial basis function network is presented in this article, to address the attitude tracking control problem of the three degree-of-freedom four-rotor hover system. The proposed controller provides convergence of system states in a pre-determined finite time and estimates the unmodeled dynamics of the four-rotor system. Dynamic model of the four-rotor system is derived with Newton's force equations. The unknown dynamics of four-rotor systems are estimated using Radial basis function. The bio-inspired global fast terminal sliding mode controller is proposed to provide chattering free finite time error convergence and to provide optimal tracking of the attitude angles while being subjected to unknown dynamics. The global stability proof of the designed controller is provided on the basis of Lyapunov stability theorem. The proposed controller is validated by (i) conducting an experiment through implementing it on the laboratory-based hover system, and (ii) through simulations. Performance of the proposed control scheme is also compared with classical and intelligent controllers. The performance comparison exhibits that the designed controller has quick transient response and improved chattering free steady state performance. The proposed bio-inspired global fast terminal sliding mode controller offers improved estimation and better tracking performance than the traditional controllers. In addition, the proposed controller is computationally cost effective and can be implanted on multirotor unmanned air vehicles with limited computational processing capabilities.

Keywords: Bio-inspired global fast terminal sliding mode controller, attitude tracking, radial basis function network, four-rotor hover system.

[1] School of Automation, Northwestern Polytechnical University, Xi'an, 710072, China.

[2] Department of Computer Science, National University of Computer and Emerging Sciences, Islamabad, 44000, Pakistan.

[3] Department for Management of Science and Technology Development, Ton Duc Thang University, Ho Chi Minh City, Vietnam.

[4] Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Vietnam.

[5] Institute of Structural Mechanics, Bauhaus University Weimar, 99423, Weimar, Germany.

[*] Corresponding Author: Shahaboddin Shamshirband. Email: shahaboddin.shamshirband@tdt.edu.vn.

1 Introduction

A noticeable surge has been witnessed in use of Multirotor UAV (MUAV) for numerous civil and strategic defense applications including aerial shooting, transportation, urban surveillance, and rescue operations, in the recent past. It is because of several outstanding properties of MUAV(s) such as its low development cost, structural simplicity, hovering capability, and low speed maneuverability [Raptis and Valavanis (2011)]. Literature shows several configurations of MUAV(s) i.e. four-rotor, quadcopter, hexarotor, and octorotor for which various controllers have been designed to tackle variable operating conditions (e.g. [Bouabdallah, Noth, Siegwan et al. (2004); Hoffmann, Huang, Waslander et al. (2007); Pounds, Mahony and Corke (2010); Rinaldi, Chiesa and Quagliotti (2012); Chen and Huzmezan (2003); Amin and Aijun (2017)]). Conventional linear controllers (e.g. PID, Linear Quadratic, and robust H∞ controllers) implemented to control MUAV(s) (e.g. [Bouabdallah, Noth, Siegwan et al. (2004); Hoffmann, Huang, Waslander et al. (2007); Pounds, Mahony and Corke (2010); Rinaldi, Chiesa and Quagliotti (2012); Chen and Huzmezan (2003); Amin and Aijun (2017); Amin and Aijun (2016)]) are designed on the mathematical model linearized about the fixed heave state. These classical linear control methods were not found practically viable for controlling trajectory tracking of MUAV, especially in presence of parametric uncertainties and modeling inaccuracies. Hence, it can be concluded that apart of the availability of several controllers for MUAV [Amin, Aijun and Shamshirband (2016)], the effective control of MUAV(s) is still active control problem due to its complex nature and highly coupled system dynamics.

Literature study reveals that nonlinear control techniques have outperformed classical control methods in dealing with parametric and un-parametric uncertainties and outward disturbances. Among others, Sliding Mode Controller (SMC) has been proved as the most efficient and effective robust technique in handling uncertain systems [Lantos and Márton (2011)]. Moreover, literature review also states several SMC based techniques developed for 3 DOF four-rotor system and quadcopter such as use of SMC for quadcopter control was introduced by Bouabdallah and Siegwart [Bouabdallah and Siegwart (2005)]. The results of the aforementioned technique exhibited poor performance and chattering in output states [Bouabdallah and Siegwart (2005)]. Xu et al. also employed SMC for quadcopter but the major limitation was that simulations were not performed on a real platform [Xu and Ozguner (2006)]. Exact feedback linearization was proposed for quadrotor control by Mian et al. [Mian and Wang (2008)], and Voos [Voos (2009)]. On comparison between feedback linearization controller and SMC, the latter showed efficient performance in noisy conditions [Lee, Kim and Sastry (2009)]. Bouadi proposed SMC based adaptive tracking controller for quadcopter with model uncertainties and in presence of external turbulences [Bouadi, Simoes Cunha, Drouin et al. (2011)]. Yang et al. presented Adaptive Fuzzy SMC (AFSMC) for attitude and position tracking of quadcopter subjected to actuator failure [Yang, Jiang and Zhang (2014)]. Recently, Sumantri et al. [Sumantri, Uchiyama and Sano (2016)] proposed a least-square based SMC for chattering free quadrotor control. However, sliding surface was usually a linear subspace of the system states in the aforementioned sliding mode based controllers. This results in the convergence of errors to origin in infinite time that only ensures asymptotic stability.

Terminal SMC (TSMC) and fast terminal SMC (FTSMC) are the two variants of SMC designed to achieve stability in finite time. Venkataraman & Gulati introduced TSMC for nonlinear systems [Venkataraman and Gulati (1992)] which was then successfully employed for uncertain linear and MIMO systems (e.g. [Man and Yu (1997); Wu, Yu, and Man (1998)]). However, TSMC proved inefficient in finite time convergence in such operational conditions where system states are far-off from the origin. Moreover, TSMC suffers from singularity problem in robotic manipulators control. The singularity problem was addressed by Feng et al. by introducing nonsingular TSMC (NTSMC) for control of rigid manipulators [Feng, Yu and Man (2002)]. The first issue was resolved by Yu & Man, by introducing FTSMC for nonlinear dynamic systems for convergence of system states, whether being close or away in fast finite time [Yu and Man (2002)]. However, FTSMC also has singularity problem [Feng, Yu and Man (2002)] which was addressed by Yu et al. [Yu, Du, Yu et al. (2008)], by introducing a nonsingular FTSMC (NFTSMC); a recursive FTSMC for finite time convergence of nth-order system to resolve singularity problem. Nonetheless, FTSMC and NFTSMC both lack adaptability and learning properties and face chattering problem [Li, Dou and Su (2011)]. Therefore, it can be observed that system stability can be compromised with poor system performance. However, augmentation of an adaptive or learning based controller with FTSMC and NFTSMC and a bio-inspired switching law can serve purpose.

With the advent of intelligent controllers, fuzzy logic and neural network-based controllers are gradually being employed in many applications and systems with uncertain system dynamics (e.g. [Yu, Li and Li (2011); Fu, Wu, Ko et al. (2011); Lian (2014); Kaiser, Chowdhury, Mamun et al. (2016); Sun and Li (2015); Chohra, Benmehrez and Farah (1998)]). In particular, Radial Basis Function Network (RBFN) controllers have been used in several mechanical and electro-mechanical systems (e.g. [Liu (2012); Feng, Xiao, Leung et al. (2014)]). An RBFN approximates the dynamical model and provides a unique solution for a specific data set. In addition, RBFN also provides noise tolerance, enhanced system stability, and quick learning as compared with other learning-based controllers [Yu and Xie (2011)]. Literature shows that a considerable number of studies have been conducted to cater parametric uncertainties, external disturbances, and actuator failure for different configurations of MUAV. However, attitude tracking problem with unmodeled dynamics approximation still needs attention, which can cause the MUAV instability during operation making it prone to intermittent issues. Recently, Amin et al. [Amin, Aijun, Khan et al. (2016)] proposed tracking controller using adaptive extended normalized radial basis function (AENRBF) to address the tracking problem with unmodeled dynamics estimation. The proposed controller was successfully tested on a four-rotor test bench system subjected to outward disturbances. However, the controller did not provide any means to control convergence time and high computation was required for the computation of ENRBF. This shows that the proposed controller is not feasible for micro MUAV aimed to operate in disturbance free environment with limited processing power. This provides the motivation to explore more on intelligent controllers for MUAV control.

In this study, a Bio-Inspired Global Fast-Terminal SMC (BIGFTSMC) in conjunction with RBFN is proposed for attitude tracking control of four-rotor test bench system. The RBFN is employed to estimate the neglected system dynamics and to overcome the

aforementioned FTSMC issues. BIGFTSMC provides fast and finite time tracking of attitude angles irrespective of system states' position from the equilibrium point. The Bio-inspired switching law is employed to eradicate chattering phenomena. The closed-loop system stability is analyzed using Lyapunov stability theorem and it is showed that the designed controller provides stable tracking performance. The efficiency of the designed controller is examined using numerical simulation and simulation results are also compared with classical PID controller and AENRBFN controller. Moreover, the proposed BIGFTSMC is tested on four-rotor test bench system to validate effectiveness through experiment. The contributions of this study are stated as:

1. BIGFTSMC in conjunction with RBFN is used for tracking control problem of MUAV which is computationally cost effective as compared with the previous work such as AENRBFN (e.g. [Amin and Aijun (2017); Amin, Aijun, Khan et al. (2016)]).
2. Fast finite time convergence of states is provided irrespective of the initial conditions, that is an exceptional feature of the designed controller unavailable in the previous work (AENRBFN) and classical controller i.e. PID (e.g. [Amin and Aijun (2017) ; Amin, Aijun, Khan et al. (2016)]).
3. Bio-inspired switching law is introduced with FTSMC for chattering free response, which is not a characteristic of AENRBFN and PID controllers.
4. The stability of the complete system with designed BIGFTSMC is investigated and proof of asymptotic stability of the closed loop system is also provided.

The rest of the article is structured as follows. In Section 2, the four-rotor test bench experimental setup, dynamic model and problem statement are presented. In Section 3, the RBFN based dynamic estimation, BIGFTSMC design, and closed-loop system stability analysis is provided. Numerical simulation and experimental findings are given in Section 4. In the end, Section 5 concludes this article.

2 Experimental setup

2.1 Structure details

The four-rotor system is a 3-DOF laboratory test bench system developed by Googol Systems [Googol Technology (2012)]. The four-rotor hover system is used to design and implement flight control laws for MUAV(s). The flight controllers designed for the four-rotor test bench can also use for attitude tracking control of other variants of aerial vehicles. The four-rotor experimental system is shown in Fig. 1.

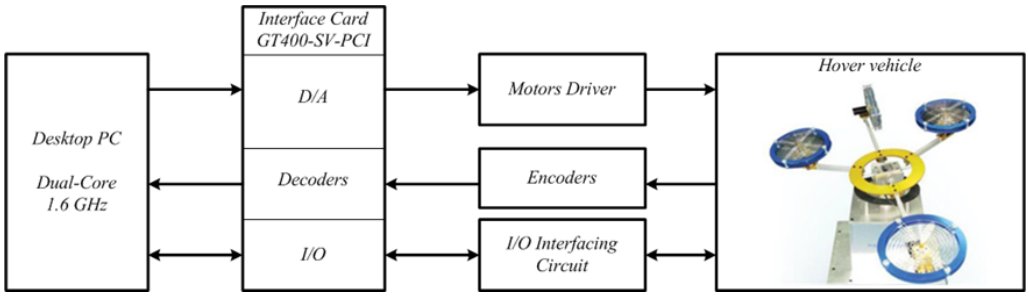

Figure 1: Four-rotor test bench experimental system

The test bench system comprises of a moving platform and a control cabinet. The moving sub-system comprises of four propellers connected to a circular gimbal called front, back, left, and right propellers. The gimbal is fixed on a universal joint that restricts the linear motion (i.e. surge, sway, and heave) of the hover system and only allows the 3DOF rotational motion (i.e. pitch, roll, and yaw). The 3DOF rotational motion is caused by the torque(s) generated by propellers. For instance, the pitch motion is caused by front, right, and left propellers; the roll motion is caused by right and left propellers; and the yaw motion is caused by the back propeller. High precision encoders are attached with the propellers' motors that determine the angles of rotational motion. The hover system moving platform is connected to a computer system through a control cabinet. The computer system provides the users with the facility to design and test flight controllers using MATLAB/Simulink. The computer system is connected to the control cabinet through a PCI card (GT400-SV-PCI). The PCI card comprises of D/A converters (DACs), decoders, digital, and analog inputs/outputs (I/Os).

2.2 Dynamical model

Four-rotor hover system has four inputs and three outputs. The thrust forces produced by the propellers denoted as F_l for left propeller, F_r for right propeller, F_f for front propeller, and F_b for back propeller are the inputs. The system outputs are denoted as $\eta = [\phi,\ \theta,\ \psi]$ where ϕ represents roll angle, θ is pitch angle and ψ is yaw angle. Front, right, and left propellers are at equal distance equidistant l_f from center as presented in Fig. 2 [Amin, Aijun, Khan et al. (2016)].

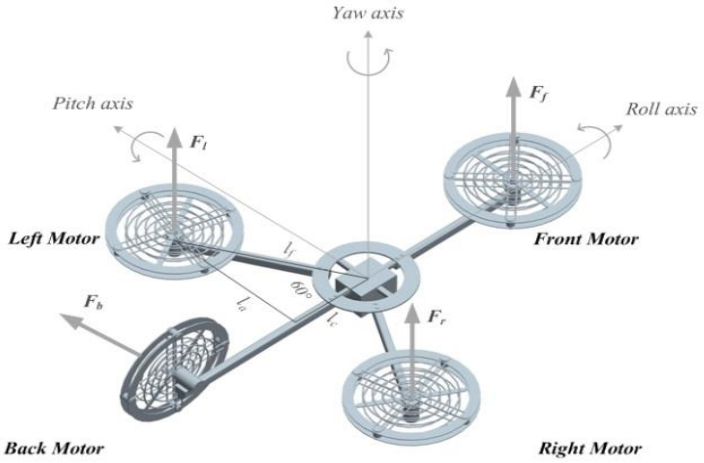

Figure 2: Four-rotor test bench system structure

The assumptions made while modeling four-rotor test bench system are stated below:

- The four-rotor system has a symmetrical and rigid structure.
- The inertia matrix of the four-rotor system is time invariant.
- Roll and pitch angle movements are limited to ±15° due to structural constraints.

The roll motion is determined using the following equation

$$J_r\ddot{\phi} = F_l l_a - F_r l_a$$

$$J_r\ddot{\phi} = K_{fc}V_l l_a - K_{fc}V_r l_a \tag{1}$$

Where J_r is the roll axis inertial moment and K_{fc} is a constant that represents voltage to thrust ratio. Fig. 2 shows that $l_a = l_f \sin 60 = \frac{\sqrt{3}}{2} l_f$. Therefore the Eq. (1) is rewritten as

$$\ddot{\phi} = \frac{\sqrt{3}}{2}\frac{K_{fcl} l_f}{J_r}\left(V_l - V_r\right) \tag{2}$$

Likewise, the pitch movement is determined by the following equation

$$J_p\ddot{\theta} = -F_f l_f - F_l l_c - F_r l_c$$

$$J_p\ddot{\theta} = -K_{fc}V_f l_f - K_{fc}V_l l_c - K_{fc}V_r l_c \tag{3}$$

Where J_p is the pitch axis inertial moment. Fig. 2 shows that $l_c = l_f \cos 60 = \frac{1}{2} l_f$. Therefore the Eq. (3) can be rewritten as

$$\ddot{\theta} = -\frac{1}{2}\frac{K_{fc} l_f}{J_p}\left(2V_f + V_l + V_r\right) \tag{4}$$

The yaw movement is determined by the following equation

$$J_y \ddot{\psi} = F_b l_f = K_{fc} V_b l_f$$

$$\ddot{\psi} = \frac{K_{fc} l_f}{J_y} V_b \tag{5}$$

Where J_y is the yaw axis inertial moment.

Let system states defined as $\eta_1 = [\phi, \theta, \psi]$ $\eta_2 = \left[\dot{\phi}, \dot{\theta}, \dot{\psi}\right]$ and control input defined as $u = \left[V_f, V_l, V_r, V_b\right]^T$. Using the above defined vectors, Eqs. (2), (4), and (5) can be rewritten as

$$\begin{bmatrix} \dot{\boldsymbol{\eta}}_1 \\ \dot{\boldsymbol{\eta}}_2 \end{bmatrix} = \begin{bmatrix} 0_{3\times3} & \boldsymbol{I}_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix} \begin{bmatrix} \boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \end{bmatrix} + \begin{bmatrix} 0_{3\times4} \\ \boldsymbol{B}_{3\times4} \end{bmatrix} \boldsymbol{u} \tag{6}$$

Where $B = \begin{bmatrix} 0 & b_1 & -b_1 & 0 \\ b_2 & b_3 & b_3 & 0 \\ 0 & 0 & 0 & b_4 \end{bmatrix}$

$$b_1 = \frac{0.866 K_{fc} l_f}{J_r},\ b_2 = -\frac{K_{fc} l_f}{J_p},\ b_3 = -\frac{0.5 K_{fc} l_f}{J_p},\ b_4 = \frac{K_{fc} l_f}{J_y}$$

2.3 Problem formulation

The complete dynamical model of four-rotor system is defined as follows

$$\ddot{\boldsymbol{\eta}} = \boldsymbol{f}\left(\boldsymbol{\eta}, \dot{\boldsymbol{\eta}}\right) + \boldsymbol{B}\boldsymbol{u} \tag{7}$$

Where $\boldsymbol{f}\left(\boldsymbol{\eta}, \dot{\boldsymbol{\eta}}\right)$ is an unknown nonlinear function that denotes neglected dynamics and *B* is a constant matrix. For the reference output state vector $\boldsymbol{\eta}_d$, the error vector is stated as

$$\boldsymbol{\xi} = \boldsymbol{\eta}_d - \boldsymbol{\eta} \tag{8}$$

This study is aimed to estimate the neglected dynamics and to propose a controller for the four-rotor hover system such that output vector η track the reference vector η_d within finite time and without any steady-state error. The assumptions made in the control design of four-rotor system are given as following:

Assumption 1: The reference state vector η_d and its time derivatives are bounded and continuous.

Assumption 2: The nonlinear function $\boldsymbol{f}\left(\boldsymbol{\eta},\dot{\boldsymbol{\eta}}\right)$ is bounded and differentiable.

3 Controller design

3.1 RBFN based dynamics estimation

In this research, the unmodeled dynamics of four-rotor test bench system are estimated using RBFN. The RBFN has feed forward network structure with four layers: (i) An input layer having *n* inputs, (ii) A hidden layer having *l* neurons, (iii) A normalized layer having *l* nodes, and (iv) An output layer, as shown in Fig. 3.

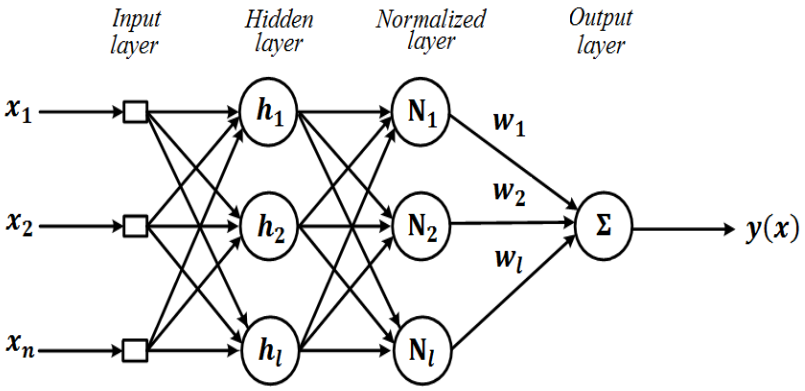

Figure 3: The RBFN architecture

Input Layer: The input layer takes an input vector $\boldsymbol{x} \in \mathfrak{R}^n$ and passes on the input vector to the hidden layer. The input vector is comprised of error and error derivatives of all output states i.e. roll, pitch, and yaw. The error values are real numbers and error ranges are same for roll and pitch but are different for yaw. Hence, input vector is not normalized between [0, 1].

Hidden Layer: The hidden layer comprises of *l* neurons and each neuron node in the hidden layer receives an input vector $\boldsymbol{x}$. Each node is comprised of a Gaussian function that changes the input to a nonlinear function, described as

$$h_j\left(x\right) = exp\left(-\frac{x - c_j^{\,2}}{2\sigma_j^{\,2}}\right), \text{for } j = 1,2,\ldots,l \tag{9}$$

Where c_j represents origin vector, $\left\|\mathrm{x} - \mathrm{c}_j\right\|^{??}$ represents Euclidean distance, and σ is the distance between input and origin of the Gaussian function. The hidden layer transmits the data to the normalized layer.

Normalized Layer: In this layer, Gaussian functions are normalized to enhance the consistency of the input data range. The output of this layer is shown as

$$N_j\left(x\right) = \frac{h_j\left(x\right)}{\sum_{j=1}^{l} h_j\left(x\right)}, \text{ for } j = 1,2,\ldots,l \tag{10}$$

Output Layer: The output layer is an algebraic summation of weighted normalized outputs, which is shown as

$$y(x) = \sum_{j=1}^{l} w_j N_j(x) \tag{11}$$

The optimization of the weight vector *w* is performed using Levenberg-Marquardt (LM) algorithm as a learning method. The cost function of RBFN is defined as

$$E = \frac{1}{2}\sum_{i=1}^{q}\left(y_{di} - y_i\right)^2 = \frac{1}{2}\sum_{i=1}^{q} e_i^{\,2}$$

Where y_{di} and y_i represents the expected response and actual response of the particular system output respectively. In LM algorithm, the weight update Δw_i is determined to minimize the following expression.

$$\Delta w_i = -\left[\boldsymbol{J}^T(w_i)\boldsymbol{J}(w_i) + \mu \boldsymbol{I}\right]^{-1} \nabla E(w_i)$$

Where w_i is the weight vector, $e(w_i)$ is the error vector, $J(w_i)$ is the Jacobian matrix of $e(w_i)$, and μ is the learning parameter. The LM algorithm continues to update weight matrix till the cost function is optimized. If the cost function is optimized at step *n*, the weight update of RBFN is given as

$$\Delta w_i = -\sum_{i=1}^{n}\left[\boldsymbol{J}^T(w_i)\boldsymbol{J}(w_i) + \mu \boldsymbol{I}\right]^{-1} \boldsymbol{J}^T(w_i) e(w_i) \tag{12}$$

The steps used in RBFN training using LMA are given in algorithm 1.

Definition 1 [Beirami (2006)]: Any continuous, bounded, and differentiable function can be defined as:

$$\boldsymbol{f}(x) = \boldsymbol{N}(\boldsymbol{x})\boldsymbol{w} + \boldsymbol{\chi} \tag{13}$$

Where $x \in \mathfrak{R}^n$ and $\boldsymbol{\chi}$ is the estimation error with a known limit $\boldsymbol{\chi_m}$ so that $\|\boldsymbol{\chi}\| \le \boldsymbol{\chi_m}$.

Remark 1: The size of a RBFN is characterized by the number of neurons in the hidden layer. In control applications, selection of hidden layer neurons is the most critical task. If number of neurons is limited, it may not fully solve the problem. On the contrary, a complex network is always inclined to memorize unnecessary data. This result in the form of noise added in the learning stage.

The selection of hidden layer neurons in control application is still a challenging task as no analytical solution has been devised till date. The typical heuristic approach is used for selection, in which various networks are trained with increasing complexity and estimation errors of all networks are observed. After training, all the network models are compared and, the network with the lowest error is selected.

Remark 2: According to Definition 1, any unknown bounded function can be estimated using RBFN with desired accuracy. The Eq. (13) can also be written as

$$\boldsymbol{f}(\boldsymbol{x}) = \hat{\boldsymbol{f}}(\boldsymbol{x}) + \tilde{\boldsymbol{f}}(\boldsymbol{x}) = \boldsymbol{N}(\boldsymbol{x})\hat{\boldsymbol{w}} + \boldsymbol{N}(\boldsymbol{x})\tilde{\boldsymbol{w}} \tag{14}$$

Where $\hat{\boldsymbol{f}}(\boldsymbol{x})$ and $\tilde{\boldsymbol{f}}(\boldsymbol{x})$ represent estimated dynamics and estimation error, respectively. Also $\hat{\boldsymbol{w}}$ and $\tilde{\boldsymbol{w}}$ denote estimated weight and error matrices with following properties [Macnab (1999)]:

1. $\boldsymbol{w} = \hat{\boldsymbol{w}} + \tilde{\boldsymbol{w}}$
2. $\hat{\boldsymbol{w}}$ and $\tilde{\boldsymbol{w}}$ are symmetrical matrices (i.e. $\hat{\boldsymbol{w}} = \hat{\boldsymbol{w}}^T$, $\tilde{\boldsymbol{w}} = \tilde{\boldsymbol{w}}^T$).
3. The derivative of matrix $\boldsymbol{w}$ is very small and $\dot{\hat{\boldsymbol{w}}} = -\dot{\tilde{\boldsymbol{w}}}$

The RBFN output in Eq. (11) is rewritten as

$$\hat{\boldsymbol{f}}(\boldsymbol{x}) = \boldsymbol{N}(\boldsymbol{x})\hat{\boldsymbol{w}} \tag{15}$$

Algorithm 1: RBFN training algorithm using LMA

1. Parameters c, λ and σ are initialized. Weights are initialized randomly between $[-a, +a]$ where $a > 0$.
2. Input vector $\boldsymbol{x}$ is received.
3. Gaussian basis functions $h(x)$ are calculated for each node in hidden layer using Eq. (9).
4. Each node in normalized layer is normalized using Eq. (10).
5. The normalized outputs are multiplied with weights to calculate output using Eq. (11).
6. The error vector and cost function are calculated as $e = y_d - y$ and $E = \frac{1}{2}e^2$ respectively where y is system output and y_d is desired output.
7. The Jacobian matrix J of error is calculated and weight update is calculated as $\Delta w = -\left[\boldsymbol{J}^T\boldsymbol{J} + \mu\boldsymbol{I}\right]^{-1}\boldsymbol{J}^T e$
8. The weight vector is updated as $w_{i+1} = w_i + \Delta w$
9. Steps 5-8 are repeated for the specific input vector $\boldsymbol{x}$ until error is within allowable range.
10. Step 2-9 for all input vectors in training data.

3.2 Bio-inspired global fast terminal sliding mode control (BIGFTSMC) design

The dynamical model of four-rotor system presented in Eq. (7) (Section 2.3) can also defined as

$$\ddot{\boldsymbol{\eta}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{B}\boldsymbol{u} \tag{16}$$

Where $\boldsymbol{f}(\boldsymbol{\eta},\dot{\boldsymbol{\eta}}) = \boldsymbol{f}(\boldsymbol{x})$ shows unknown dynamics approximated using RBFN. Using Eq. (14), Eq. (16) is rewritten as

$$\ddot{\boldsymbol{\eta}} = \hat{\boldsymbol{f}}(\boldsymbol{x}) + \tilde{\boldsymbol{f}}(\boldsymbol{x}) + \boldsymbol{B}\boldsymbol{u} \tag{17}$$

The global fast terminal sliding vector is proposed as

$$\boldsymbol{s} = \dot{\boldsymbol{\xi}} + \boldsymbol{\lambda}\boldsymbol{\xi} + \beta\boldsymbol{\xi}^{\gamma} \tag{18}$$

Where $\boldsymbol{s} = diag\left[s_1, s_{2,} s_3\right]$, $\beta > 0$, $\gamma = \frac{q}{p}$ such that *p* and *q* are odd integers $(p > q > 0)$, and $\boldsymbol{\lambda} = diag\left[\lambda_1, \lambda_{2,} \lambda_3\right]$ where $(\lambda_i > 0, \text{ for } i = 1,2,3)$.

The time derivative of Eq. (18) results in

$$\dot{\boldsymbol{s}} = \ddot{\boldsymbol{\xi}} + \boldsymbol{\lambda}\dot{\boldsymbol{\xi}} + \beta\frac{d}{dt}\boldsymbol{\xi}^{\gamma}$$

$$\dot{\boldsymbol{s}} = \ddot{\boldsymbol{\eta}}_{\boldsymbol{d}} - \ddot{\boldsymbol{\eta}} + \boldsymbol{\lambda}\dot{\boldsymbol{\xi}} + \beta\frac{d}{dt}\boldsymbol{\xi}^{\gamma} \tag{19}$$

Substitution of Eq. (17) in Eq. (19) gives

$$\dot{\boldsymbol{s}} = \ddot{\boldsymbol{\eta}}_{\boldsymbol{d}} - \hat{\boldsymbol{f}}(\boldsymbol{x}) - \tilde{\boldsymbol{f}}(\boldsymbol{x}) - \boldsymbol{B}\boldsymbol{u} + \boldsymbol{\lambda}\dot{\boldsymbol{\xi}} + \beta\frac{d}{dt}\boldsymbol{\xi}^{\gamma} \tag{20}$$

Theorem 1: The tracking error $\boldsymbol{\xi}$ of attitude angles of the four-rotor system will exponentially converge to origin in finite time if there exists a control law defined as

$$\boldsymbol{u} = -\boldsymbol{B}^{-\dagger}\left[\hat{\boldsymbol{f}}(\boldsymbol{x}) - \ddot{\boldsymbol{\eta}}_{\boldsymbol{d}} - \boldsymbol{\lambda}\dot{\boldsymbol{\xi}} - \beta\frac{d}{dt}\boldsymbol{\xi}^{\gamma} + \left(\varphi\boldsymbol{s} + \alpha\boldsymbol{s}^{\gamma}\right)\boldsymbol{\delta}_{\boldsymbol{s}}\right] \tag{21}$$

Where $\boldsymbol{B}^{-\dagger}$ represents the pseudo inverse of matrix $\boldsymbol{B}$, $\varphi, \alpha > 0$, and $\boldsymbol{\delta}_{\boldsymbol{s}}$ is the bio-inspired switching law.

Proof: First we provide the proof of tracking error convergence to the zero with the defined sliding surface and then the proof of finite time conversion is provided. Substitution of the proposed control law (Eq. (21)) in Eq. (20) results in

$$\dot{\boldsymbol{s}} = -\tilde{\boldsymbol{f}}(\boldsymbol{x}) - \left(\varphi\boldsymbol{s} + \alpha\boldsymbol{s}^{\gamma}\right)\boldsymbol{\delta}_{\boldsymbol{s}} \tag{22}$$

Multiplying s with the Eq. (22) results in

$$\boldsymbol{s}\dot{\boldsymbol{s}} = -\boldsymbol{s}\,\tilde{\boldsymbol{f}}(\boldsymbol{x}) - \left(\varphi\boldsymbol{s}^2 + \alpha\boldsymbol{s}^{\gamma+1}\right)\boldsymbol{\delta}_{\boldsymbol{s}} \tag{23}$$

Since γ, α and φ are positive numbers and $\boldsymbol{\delta}_{\boldsymbol{s}}$ is a switching law, this implies

$$\varphi\boldsymbol{s}^2 > 0 \Rightarrow -\varphi\boldsymbol{s}^2 < 0$$

In addition, $-\boldsymbol{s}\,\tilde{\boldsymbol{f}}(\boldsymbol{x}) - \alpha\boldsymbol{s}^{\gamma+1} = -s\left(\tilde{\boldsymbol{f}}(\boldsymbol{x}) + \alpha\boldsymbol{s}^{\gamma}\right)$

where $\tilde{\boldsymbol{f}}(\boldsymbol{x})+\alpha \boldsymbol{s}^{\gamma}>0$ therefore $-s\left(\tilde{\boldsymbol{f}}(\boldsymbol{x})+\alpha \boldsymbol{s}^{\gamma}\right)<0$, which subsequently gives $\alpha>\left|\frac{\tilde{\boldsymbol{f}}(\boldsymbol{x})}{\boldsymbol{s}^{\gamma}}\right|$. According to definition 1, $\tilde{\boldsymbol{f}}(\boldsymbol{x})$ is bounded, therefore we have $\alpha \geq\left|\frac{\tilde{\boldsymbol{f}}(\boldsymbol{x})_{\max }}{\boldsymbol{s}^{\gamma}}\right|$. It is concluded that $\boldsymbol{s}\dot{\boldsymbol{s}} \leq 0$ and hence showed that sliding surface and its differential converges to zero.

The sliding surface defined in Eq. (18) at $\boldsymbol{s}=0$ can be written as

$$\dot{\xi}=-\lambda \xi-\beta \xi^{\gamma} \tag{24}$$

When the tracking error ξ is far-off from the origin, the convergent time is dominated by the fast terminal term $\beta \xi^{\gamma}$. When the error approaches the sliding surface the convergent time is determined using $\dot{\xi}=-\lambda \xi$. Eq. (24) can be written as

$$\frac{d \xi}{d t}=-\lambda \xi-\beta \xi^{\gamma}$$

$$\int_0^{t_s} d t=\int_{\xi_0}^0\left(\frac{1}{-\lambda \xi-\beta \xi^{\gamma}}\right) d \xi$$

Thus, the time interval in which tracking error converges to zero is given as

$$t_s=\frac{1}{\lambda(1-\gamma)} \ln \frac{\lambda \xi(0)^{1-\gamma}+\beta}{\beta} \tag{25}$$

Therefore, it is showed that by appropriate values of $\lambda, \beta,$ and γ, the tracking error is converged to origin in finite predefined time t_s .

Bio-Inspired Switching Law:

The bio-inspired switching law was introduced by Yang et al. [Yang, Zhu, Yuan et al. (2012)]. In this work a modified bio-inspired switching law $\boldsymbol{\delta_s}$ is combined with FTSMC for chattering free finite time convergence of system states. The switching law rate is defined as

$$\frac{d \boldsymbol{\delta_s}}{d x}=-k \boldsymbol{\delta_s}+\delta^{+}\left(\delta_U-\boldsymbol{\delta_s}\right)+\delta^{-}\left(\delta_L+\boldsymbol{\delta_s}\right) \tag{26}$$

Where $\boldsymbol{\delta_s}$ is the bio-inspired switching law, *k* is the decay constant, δ_U and δ_L are upper and lower bounds of the switching law $\left(\delta_U, \delta_L>0\right)$, and δ^{+} and δ^{-} are positive and negative switching functions such that

$$\delta^{+} = \begin{cases} \boldsymbol{s} & \boldsymbol{s} > 0 \\ 0 & \boldsymbol{s} < 0 \end{cases}$$

$$\delta^{-} = \begin{cases} \boldsymbol{s} & \boldsymbol{s} < 0 \\ 0 & \boldsymbol{s} > 0 \end{cases}$$

For positive sliding surface $\boldsymbol{s} > 0$, the negative switching function becomes zero and solution of Eq. (26) results in the following switching law:

$$\boldsymbol{\delta_s} = \delta_U \boldsymbol{s} e^{-(k+\boldsymbol{s})t}$$

Since $k + \boldsymbol{s} > 0$, it is clear from the above equation that as $\boldsymbol{s} \to 0$, $\boldsymbol{\delta_s} \to 0$.

Similarly, for negative sliding surface, $\boldsymbol{s} < 0$, the positive switching function becomes zero and solution of Eq. (26) results in the following switching law:

$$\boldsymbol{\delta_s} = \delta_L \boldsymbol{s} e^{-(k-\boldsymbol{s})t}$$

Since $k - \boldsymbol{s} > 0$, it is clear from the above equation that as $\boldsymbol{s} \to 0$, $\boldsymbol{\delta_s} \to 0$.

Hence, it shows that the bio-inspired switching law for different sliding surfaces results in smooth switching without any chattering.

3.3 Stability analysis

The definitions used in the stability analysis are stated here [Slotine and Li (1991)].

Definition 2: If a positive definite Lyapunov candidate function $V(x)$ exists for a dynamical system so that $V(0) = 0$ and $\dot{V}(x) < 0$, then the dynamical system is said to be asymptotically stable.

Definition 3: If asymptotic stability holds for any of the initial states, the system is said to be globally asymptotically stable.

Theorem 2: If the proposed controller (Eq. (21)) in conjunction with RBFN based estimated dynamics $\hat{\boldsymbol{f}}(\boldsymbol{x})$ is used for four-rotor system control, then global asymptotic stability of the closed-loop system is guaranteed.

Proof: Let the Lyapunov function is defined as

$$V = \frac{1}{2}\boldsymbol{s}^2 + \frac{1}{2}\Gamma \tilde{\mathbf{w}}^{\boldsymbol{T}} \tilde{\mathbf{w}} \tag{27}$$

The time derivative of the defined Lyapunov function results as

$$\dot{V} = \boldsymbol{s}\dot{\boldsymbol{s}} + \Gamma \tilde{\mathbf{w}}^{\boldsymbol{T}} \dot{\tilde{\mathbf{w}}}$$

Substituting $\boldsymbol{s}\dot{\boldsymbol{s}}$ from Eq. (23) and $\dot{\hat{\boldsymbol{w}}} = -\dot{\tilde{\boldsymbol{w}}}$ from Remark 2 results in

$$\dot{V} = -\boldsymbol{s}\,\tilde{\boldsymbol{f}}(\boldsymbol{x}) - \left(\varphi \boldsymbol{s}^2 + \alpha \boldsymbol{s}^{\gamma+1}\right) \boldsymbol{\delta_s} - \Gamma \tilde{\mathbf{w}}^{\boldsymbol{T}} \dot{\hat{\mathbf{w}}}$$

$$= -\boldsymbol{s}\tilde{\mathbf{w}}^{\boldsymbol{T}} N(\boldsymbol{x}) - \Gamma \tilde{\mathbf{w}}^{\boldsymbol{T}} \dot{\hat{\mathbf{w}}} - \left(\varphi \boldsymbol{s}^2 + \alpha \boldsymbol{s}^{\gamma+1}\right) \boldsymbol{\delta_s}$$

$$= -\tilde{\mathbf{w}}^T \left(\boldsymbol{s} N(\boldsymbol{x}) + \Gamma \dot{\hat{\mathbf{w}}} \right) - \left(\varphi \boldsymbol{s}^2 + \alpha \boldsymbol{s}^{\gamma+1} \right) \boldsymbol{\delta}_s \tag{28}$$

The weight update is taken as

$$\dot{\hat{\mathbf{w}}} = -\Gamma^{-1} \boldsymbol{s} N(\boldsymbol{x}) \tag{29}$$

Now the Eq. (28) becomes

$$\dot{V} = -\left(\varphi \boldsymbol{s}^2 + \alpha \boldsymbol{s}^{\gamma+1} \right) \boldsymbol{\delta}_s \tag{30}$$

Since γ, α and φ are positive numbers, this implies $\varphi \boldsymbol{s}^2 + \alpha \boldsymbol{s}^{\gamma+1} \geq 0$ so the Eq. (30) becomes $\dot{V} < 0$. Since the Lyapunov function is steadily decreased, therefore closed loop system is globally stable. Thus, the designed system is globally asymptotically stable.

4 Results and discussions

In this section simulation and experimental findings are shown to demonstrate the efficiency of the designed controller. The simulations and experiment are performed to validate the proposed controller and to implement the controller on the laboratory-based hover system. First of all, unit step output of BIGFTSMC is determined and compared with outputs of classical PID and AENRBFN controllers. Next, tracking response of BIGFTSMC is studied with sinusoidal input trajectory. The four-rotor test bench system parameters are listed in Tab. 1.

Table 1: Four-rotor hover system parameters

Parameter	Description	Value	Unit
m	System mass	1	Kg
l	Distance between each motor and origin	0.50	m
J_P	Pitch axis inertial moment	0.91	Kg.m^2
J_R	Roll axis inertial moment	0.41	Kg.m^2
J_Y	Yaw axis inertial moment	1.31	Kg.m^2
K_{fc}	Voltage to lift ratio	15	N/V

4.1 RBFN supervised training

The RBFN is trained to estimate the four-rotor system unknown dynamics. The proposed RBFN has three substructures with respect to system outputs. Each substructure is comprised of two input nodes, five hidden nodes, five normalized nodes, and an output node. The input vector is comprised of error and error derivatives for all output states i.e. roll, pitch, and yaw, expressed as $x(t) = \left[e_1 \quad de_1 \quad e_2 \quad de_2 \quad e_3 \quad de_3 \right]^T$. The initial weights of the hidden neurons are randomly generated within the interval [-1, 1]. The training data covers the learning space $\{-15^\circ, -10^\circ, -5^\circ, 0^\circ, 5^\circ, 10^\circ, 15^\circ\}$ considering the maneuverability of the hover system. The number of learning episodes is set to 10000 with learning rate $\mu = 0.1$. Each learning episode will stop if the hover system tracks the

desired reference input and error is within allowable range $\varepsilon = 0.05$. The heuristic approach is considered for selection of number of learning episodes i.e. RBFN is trained with various learning episodes and it is found that optimal solution can be obtained with around 10000 learning episodes.

The error convergence for the RBFN training is shown in Fig. 4.

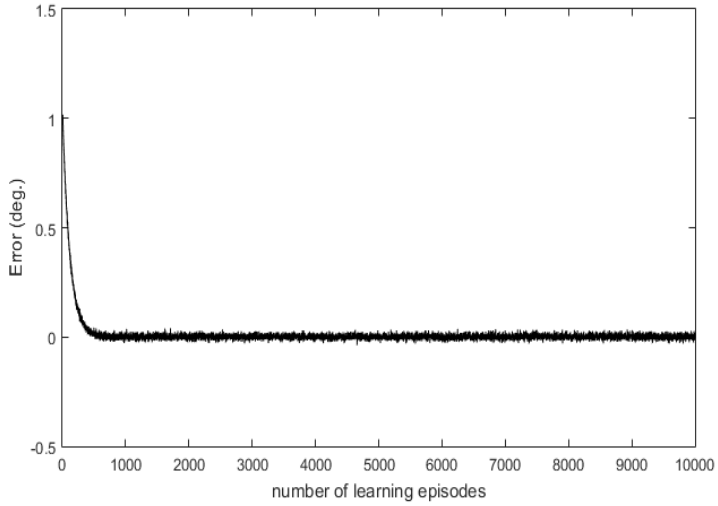

Figure 4: Error convergence in RBFN training

4.2 Simulation results

Simulation results are presented in this sub-section. Initially, step output of BIGFTSMC is determined and compared with step outputs of PID and AENRBFN controllers. In the second case, the response of four-rotor system with sinusoidal reference trajectory is presented. For simulation, frictional moment is supposed as unknown dynamics to investigate the performance of proposed RBFN. Therefore, unknown dynamics is considered as

$$\boldsymbol{f}\left(\boldsymbol{x}\right) = \mathrm{K}\,\boldsymbol{\eta} \tag{31}$$

where K represents friction constant matrix i.e. $\mathrm{K} = \mathrm{diag}\left[\mathrm{K}_{fr1}, \mathrm{K}_{fr2}, \mathrm{K}_{fr3}\right]$. After few trials, selected controller parameters are $\boldsymbol{\lambda} = diag\left[2,2,2\right]$, $\beta = 1$, $\varphi = 0.15$, $\gamma = \frac{3}{5}$, and $\alpha = \frac{\Lambda}{\left|\boldsymbol{s}^{\gamma}\right|} + \varepsilon$, where $\Lambda = 0.50$ and $\varepsilon = 0.10$. Firstly, step input of 2° is used as a reference input for both tilt angles. The tracking response of the proposed BIGFTSMC, as well as PID and AENRBFN controller responses are shown in Fig. 5. The responses clearly show that the proposed BIGFTSMC controller has an improved transient response and system's outputs follow the given reference input faster than PID and AENRBFN controllers. In addition, BIGFTSMC's response is free form chattering or oscillations which are normally present in SMC based controllers. PID controller offers slightly fast

settling time as compared to AENRFBN but the response has unwanted overshoots with constant steady state error. Error plots are presented in Fig. 6. Error plots clearly show that error states converge to equilibrium faster than AENRBFN controller, which offers better error response as compared with LQR and AFSMC [Amin, Aijun, Khan et al. (2016)]. Hence, it indicates that BIGFTSMC is capable of steady and improved tracking as compared with conventional and AENRBFN controllers. The performance comparison is summarized in Tab. 2.

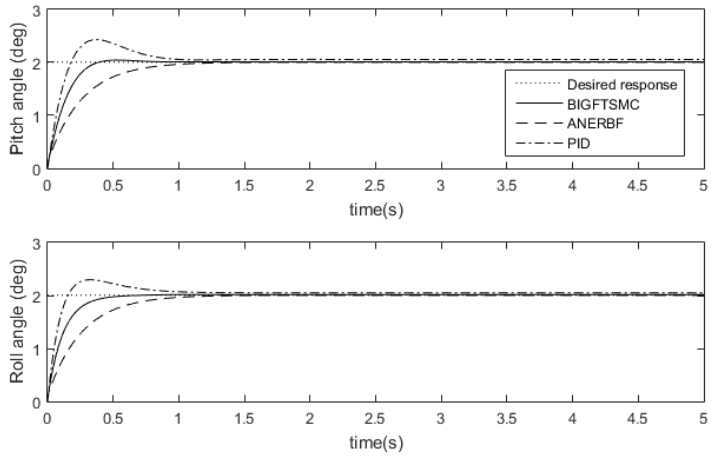

Figure 5: Step responses of PID, ANERBF and BIGFTSMC controllers

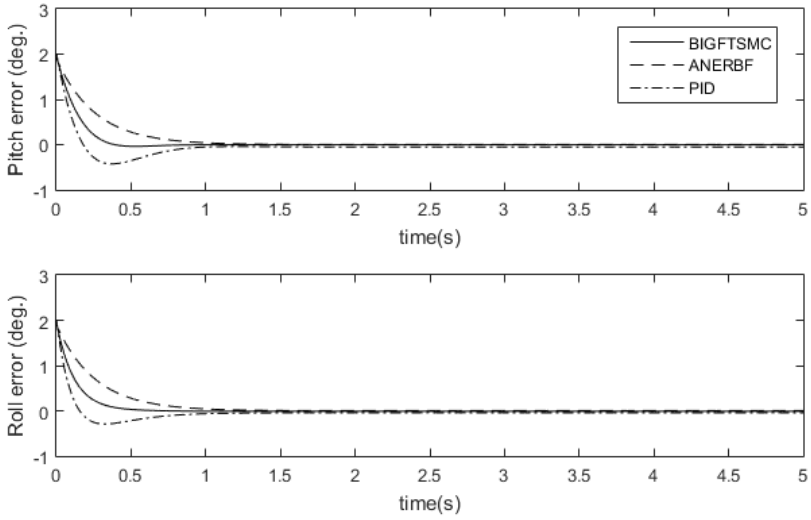

Figure 6: Error responses of PID, ANERBF and BIGFTSMC controllers

Table 2: Comparison of PID, AENRBFN and BIGFTSMC controllers

Control law	Settling time (sec)	Percent Overshoot	Steady state error
PID			
Roll	0.90	12.1	0.04
Pitch	0.90	18.2	0.06
AENRBF			
Roll	0.97	0	0
Pitch	0.98	0	0
BIGFTSMC			
Roll	0.48	0	0
Pitch	0.33	0	0

In the second case, sine wave of 3° is used as the reference inputs for roll, pitch and yaw angles to evaluate controller efficiency. The tracking outputs of this case are presented in Fig. 7. The outputs show that the designed BIGFTSMC follows the input trajectory in real-time without any observable error. Error plots are presented in Fig. 8. The plots clearly show that within no time errors are converged to less than 0.01% amplitude of the input signal and remain in that limit. This limit is quite suitable for practical applications as it is due to electromechanical nature of the system. The unknown dynamics approximation errors are presented in Fig. 9. The approximation errors are within acceptable range which is obvious from Fig. 9. In a nutshell, simulations clearly exhibit that BIGFTSMC provides an improved and better tracking response. Moreover, RBFN very quickly estimates the unknown dynamics and prevents the system to become unstable due to modeling inaccuracies.

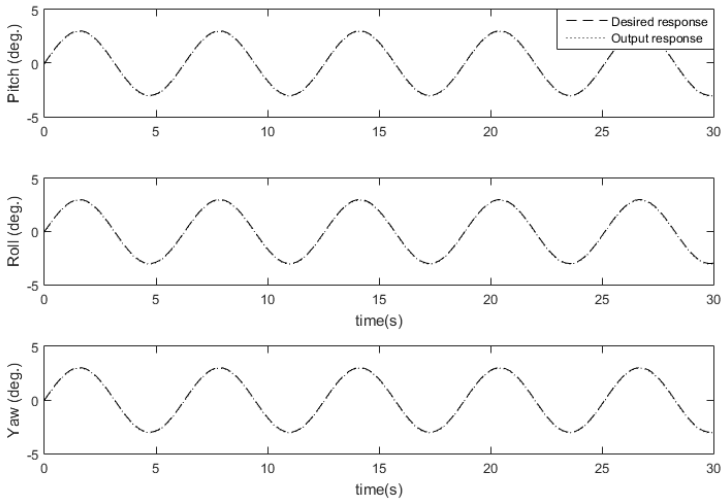

Figure 7: BIGFTSMC output response to sine inputs

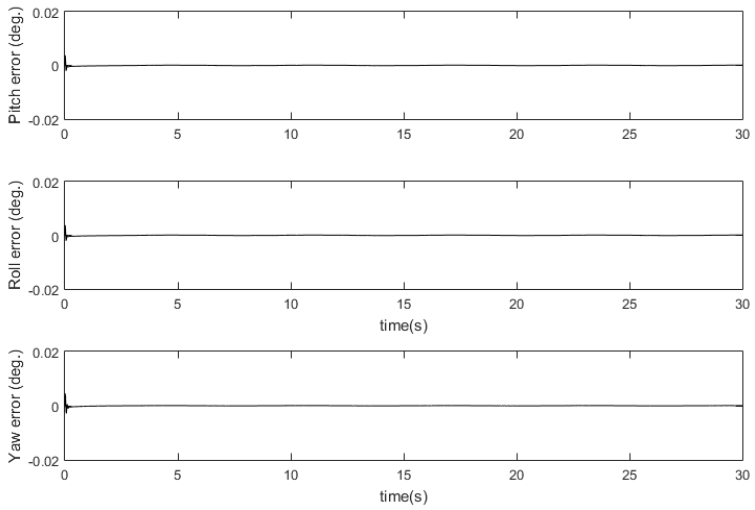

Figure 8: BIGFTSMC error response

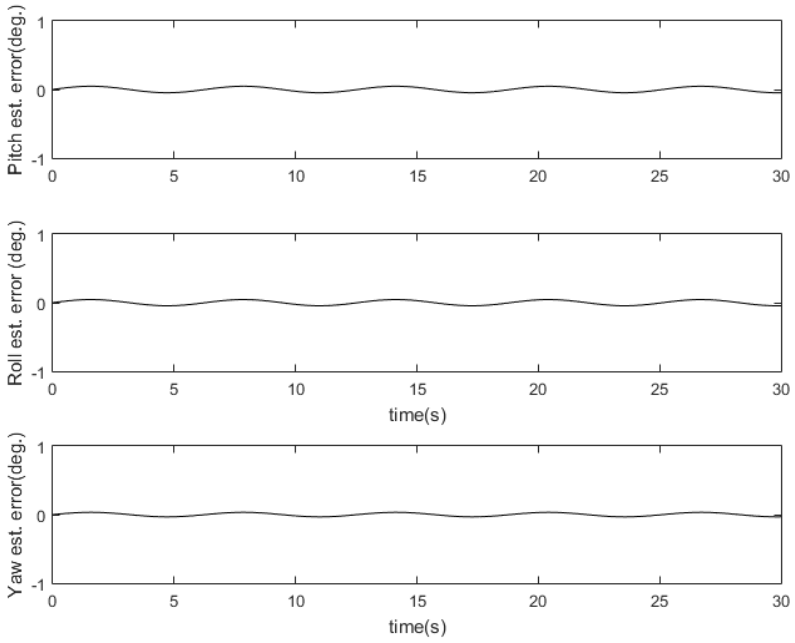

Figure 9: Unknown dynamics estimation errors

4.3 Experimental results

After achieving satisfactory simulations, the proposed control law is applied to a four-rotor hover test bench system for controller validation. The four-rotor test bench system is presented in Section 2. The computer with the test bench is available with Simulink based software interface for implementation of designed control laws. The software interface is available with LQR controller as shown in Fig. 10. The LQR block is replaced by the proposed controller defined in Eq. (21). The RBFN parameters used in

simulations are also used in the experiment. The sine waves of amplitude 3° with 180° phase shift are used as reference inputs for roll and pitch angles. The tracking outputs are presented in Fig. 11. The outputs clearly show that roll and pitch outputs track the reference signals without any error and chattering. The error plots are presented in Fig. 12. The errors magnitudes are slightly higher than that the ones obtained in simulation but are still in the acceptable error limit for real-time tracking. Unknown dynamics estimation plots are presented in Fig. 13. The plots clearly show that estimation response is consistent with the reference input. Thus, RBFN has proved very effective in real time unknown dynamics approximation. It is concluded form experimental results that the designed BIGFTSMC provides improved tracking performance. Moreover, controller performance in experimental results validates the numerical simulations.

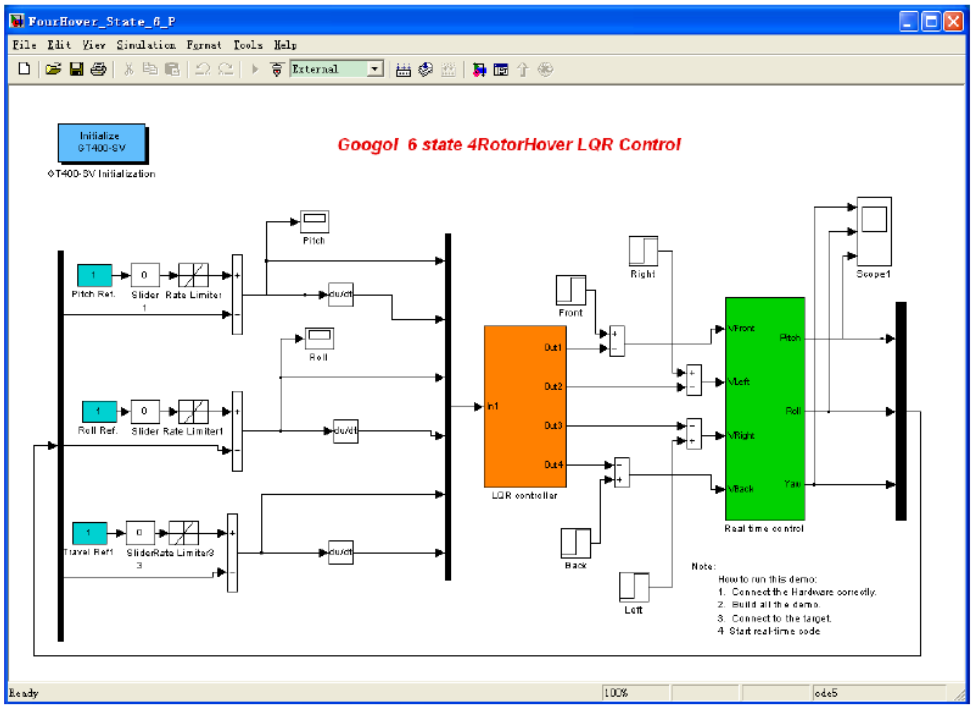

Figure 10: Four-rotor hover system controller relaization

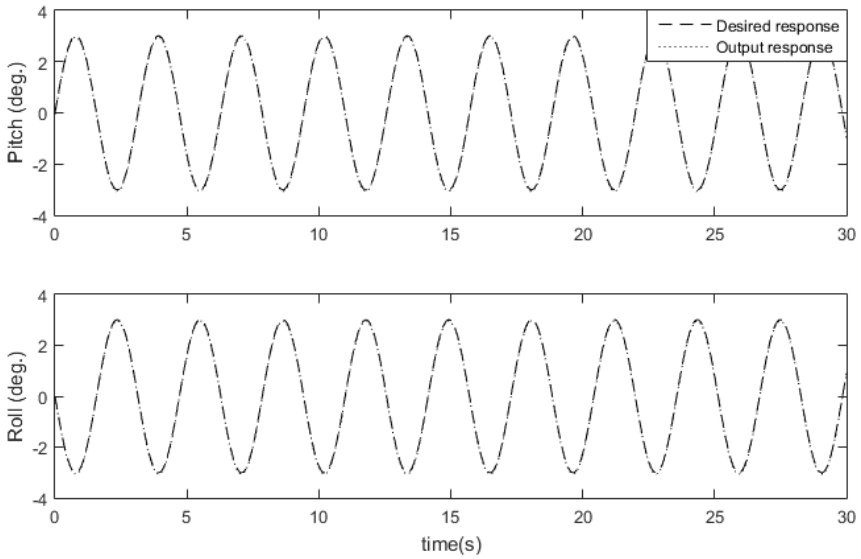

Figure 11: Four-rotor test bench system output response

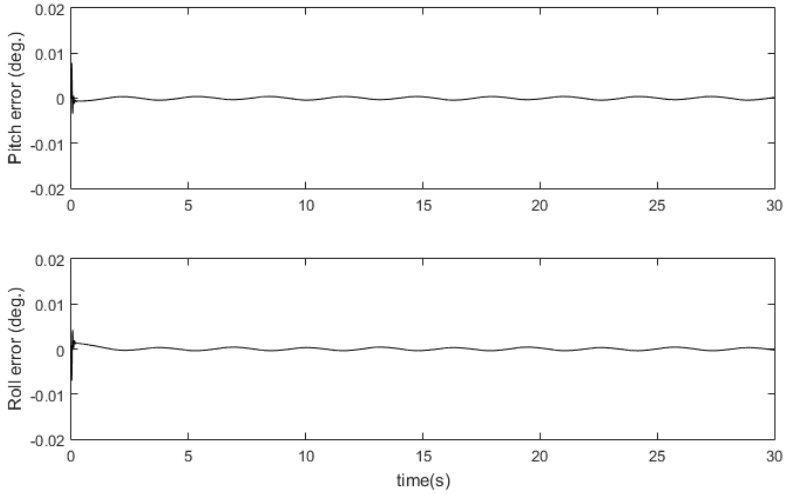

Figure 12: Four-rotor test bench system error response

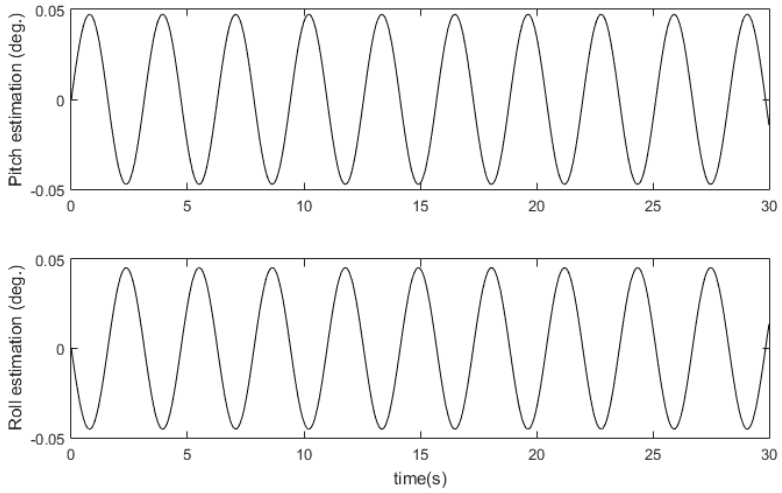

Figure 13: Four-rotor test bench system dynamics estimation response

5 Conclusion

In this work, RBFN based BIGFTSMC is presented to address tracking control problem of four-rotor test bench system. Dynamic model of the four-rotor system is derived with Newton's force equations. The unknown dynamics of four-rotor systems are estimated using RBFN and LMA is used as learning algorithm to train RBFN. The BIGFTSMC is designed using bio-inspired switching law to ensure fast convergence of errors and to provide improved chattering free tracking response even with modeling inaccuracies. The global stability proof of BIGFTSMC is provided using Lyapunov stability theorem. The comparison with PID and adaptive ENRFBN controller verifies that the proposed controller achieves improved and fast transient response and chattering free steady state response. In addition, the proposed BIGFTSMC controller is computationally cost effective and can be implanted even on micro MUAV with limited computation processing capabilities. Simulation and experimental results show acceptable estimation and tracking performance which demonstrate the effectiveness of the designed controller. This work can be combined with fault tolerant control (FTC) method to address actuator failure problem. Furthermore, disturbance observer can be designed and included to mitigate the external disturbance effect for operation in extreme weather conditions.

References

Amin, R.; Aijun, L.; Shamshirband, S. (2016): A review of quadrotor UAV : Control methodologies and performance evaluation. *International Journal of Control Automation and Systems*, vol. 10, pp. 87-103.

Amin, R.; Aijun, L. (2016): Design of mixed sensitivity H∞ control for four-rotor hover vehicle. *International Journal of Control Automation and Systems*, vol. 11, pp. 89-103.

Amin, R.; Aijun, L. (2017): Modelling and robust attitude trajectory tracking control of 3-DOF four-rotor Hover vehicle. *Aircraft Engineering and Aerospace Technology*, vol. 89, no. 1, pp. 87-98.

Amin, R.; Aijun, L.; Khan, M. U.; Shamshirband, S.; Kamsin, A. (2016): An adaptive trajectory tracking control of four-rotor hover vehicle using extended normalized radial basis function network. *Mechanical Systems and Signal Processin*, vol. 83, pp. 53-74.

Beirami, A. (2006): Direct neural-adaptive control of robotic manipulators using a forward dynamics approach. *2006 Canadian Conference on Electrical and Computer Engineering*, pp. 363-367.

Bouabdallah, S.; Noth, A.; Siegwan, R.; Siegwart, R. (2004): PID *vs.* LQ control techniques applied to an indoor micro quadrotor. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2451-2456.

Bouabdallah, S.; Siegwart, R. (2005): Backstepping and sliding-mode techniques applied to an indoor micro Quadrotor. *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 2247-2252.

Bouadi, H.; Simoes Cunha, S.; Drouin, A.; Mora-Camino, F. (2011): Adaptive sliding mode control for quadrotor attitude stabilization and altitude tracking. *IEEE 12th International Symposium on Computational Intelligence and Informatics*, pp. 449-455.

Chen, M.; Huzmezan, M. (2003): A combined MBPC/2 dof H∞ controller for a quad rotor UAV. *AIAA Guidance, Navigation, and Control Conference and Exhibit.*

Chohra, A.; Benmehrez, C.; Farah, A. (1998): Neural navigation approach for intelligent autonomous vehicles (IAV) in partially structured environments. *Applied Intelligence*, pp. 219-233.

Feng, R. B.; Xiao, Y.; Leung, C. S.; Tsang, P. W. M.; Sum, J. (2014): An improved fault-tolerant objective function and learning algorithm for training the radial basis function neural network. *Cognitive Computation*, vol. 6, pp. 293-303.

Feng, Y.; Yu, X.; Man, Z. (2002): Non-singular terminal sliding mode control of rigid manipulators. *Automatica*, vol. 38, pp. 2159-2167.

Fu, Y. Y.; Wu, C. J.; Ko, C. N.; Jeng, J. T. (2011): Radial basis function networks with hybrid learning for system identification with outliers. *Applied Soft Computing*, vol. 11, pp. 3083-3092.

Googol Technology (2012): *Four-rotor hover system, User Manual V1.2*. http://www.googoltech.com/uploads/catalog/1356/4-rotor%20hover%20vehicle.pdf.

Hoffmann, G. G.; Huang, H.; Waslander, S. L.; Tomlin, C. (2007): Quadrotor helicopter flight dynamics and control: Theory and experiment. *AIAA Guidance, Navigation, and Control Conference and Exhibit Reston*, pp. 1-20.

Kaiser, M. S.; Chowdhury, Z. I.; Mamun, A. S.; Hussain, A.; Mahmud, M. (2016): A neuro-fuzzy control system based on feature extraction of surface electromyogram signal for solar-powered wheelchair. *Cognitive Computation*, pp. 1-9.

Lantos, B.; Márton, L. (2011): *Nonlinear Control of Vehicles and Robots, Advances in Industrial Control. 1st ed.* Springer-Verlag London.

Lee, D.; Kim, H. J.; Sastry, S. (2009): Feedback linearization *vs.* adaptive sliding mode control for a quadrotor helicopter. *International Journal of Control Automation and Systems*, vol. 7, pp. 419-428.

Li, H.; Dou, L.; Su, Z. (2011): Adaptive nonsingular fast terminal sliding mode control for electromechanical actuator. *International Journal of Systems Science*, vol. 44, pp. 401-415.

Lian, R. J. (2014): Adaptive self-organizing fuzzy sliding-mode radial basis-function neural-network controller for robotic systems. *IEEE Transactions on Industrial Electronics*, vol. 61, pp. 1493-1503.

Liu, J. (2012): *Radial Basis Function (RBF) Neural Network Control for Mechanical Systems*. Tsinghua University Press.

Macnab, C. J. B. (1999): *Stable Neural-Netowrk Control of Structurally Flexible Space Manipulators (Ph.D. Thesis)*. University of Toronto.

Man, Z.; Yu, X. (1997): Terminal sliding mode control of MIMO linear systems. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 44, no. 11, pp. 1065-1070.

Mian, A. A.; Wang, D. (2008): Dynamic modeling and nonlinear control strategy for an underactuated quad rotor rotorcraft. *Journal of Zhejiang University Science A*, vol. 9, pp. 539-545.

Pounds, P.; Mahony, R.; Corke, P. (2010): Modelling and control of a large quadrotor robot. *Control Engineering Practice*, vol. 18, pp. 691-699.

Raptis, I. A.; Valavanis, K. P. (2011): *Linear and Nonlinear Control of Small-Scale Unmanned Helicopters*. Springer Dordrecht Heidelberg.

Rinaldi, F.; Chiesa, S.; Quagliotti, F. (2012): Linear quadratic control for quadrotors UAVs dynamics and formation flight. *Journal of Intelligent & Robotic Systems*, vol. 70, pp. 203-220.

Slotine, J. J.; Li, W. (1991): *Applied Nonlinear Control*. Englewood Cliffs, New Jersey: Prentice-Hall Inc.

Sumantri, B.; Uchiyama, N.; Sano, S. (2016): Least square based sliding mode control for a quad-rotor helicopter and energy saving by chattering reduction. *Mechanical Systems and Signal Processing*, vol. 66-67, pp. 769-784.

Sun, J.; Li, Z. (2015): Development and implementation of a wheeled inverted pendulum vehicle using adaptive neural control with extreme learning machines. *Cognitive Computation*, vol. 7, pp. 740-752.

Venkataraman, S.; Gulati, S. (1992): Control of nonlinear systems using terminal sliding modes. *American Control Conference*, pp. 891-893.

Voos, H. (2009): Nonlinear control of a quadrotor micro-uav using feedback-linearization. *IEEE International Conference on Mechatronics*, pp. 1-6.

Wu, Y.; Yu, X.; Man, Z. (1998): Terminal sliding mode control design for uncertain dynamic systems. *Systems & Control Letters*, vol. 34, pp. 281-287.

Xu, R.; Ozguner, U. (2006): Sliding mode control of a quadrotor helicopter. *IEEE Conference on Decision & Control*, pp. 4957-4962.

Yang, H.; Jiang, B.; Zhang, K. (2014): Direct self-repairing control for quadrotor helicopter attitude systems. *Mathematical Problems in Engineering*, no. 6, pp. 1-11.

Yang, S. X.; Zhu, A.; Yuan, G.; Meng, M. Q. H. (2012): A bioinspired neurodynamics-based approach to tracking control of mobile robots. *IEEE Transactions on Industrial Electronics*, vol. 59, no. 8, pp. 3211-3220.

Yu, H.; Xie, T. (2011): Advantages of radial basis function networks for dynamic system design. *IEEE Transactions on Industrial Electronics*, vol. 58, pp. 5438-5450.

Yu, S.; Du, J.; Yu, X.; Xu, H. (2008): A novel recursive terminal sliding mode with finite-time convergence. *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 5945-5949.

Yu, X.; Man, Z. (2002): Fast terminal sliding-mode control design for nonlinear dynamical systems. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 49, no. 2, pp. 261-264.

Yu, Z.; Li, G.; Li, X. (2011): Robust adaptive neural network tracking control for manipulators with unmodeled dynamics. *2nd International Conference on Intelligent Control and Information Processing*, vol. 2, pp. 574-578.