



**QUEEN'S
UNIVERSITY
BELFAST**

Optimal adaptive neural PI full-order sliding mode control for robust fault tolerant control of uncertain nonlinear system

Van, M., & Do, X. P. (2020). Optimal adaptive neural PI full-order sliding mode control for robust fault tolerant control of uncertain nonlinear system. *European Journal of Control*. Advance online publication. <https://doi.org/10.1016/j.ejcon.2019.12.005>

Published in:
European Journal of Control

Document Version:
Peer reviewed version

Queen's University Belfast - Research Portal:
[Link to publication record in Queen's University Belfast Research Portal](#)

Publisher rights

Copyright 2019 Elsevier Ltd.

This manuscript is distributed under a Creative Commons Attribution-NonCommercial-NoDerivs License

(<https://creativecommons.org/licenses/by-nc-nd/4.0/>), which permits distribution and reproduction for non-commercial purposes, provided the author and source are cited.

General rights

Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.

Open Access

This research has been made openly available by Queen's academics and its Open Research team. We would love to hear how access to this research benefits you. – Share your feedback with us: <http://go.qub.ac.uk/oa-feedback>

Optimal Adaptive Neural PI Full-Order Sliding Mode Control for Robust Fault Tolerant Control of Uncertain Nonlinear System

Mien Van¹

*School of Electronics, Electrical Engineering and Computer Science
Queen's University Belfast, Belfast, United Kingdom*

Xuan Phu Do

*MediRobotics Laboratory, Mechatronics and Sensor Systems Technology
Vietnamese-German University, Vietnam*

Abstract

This paper proposes a robust fault tolerant control scheme for a class of second-order uncertain nonlinear systems. First, a novel PI full-order sliding mode (PI-FOSM) control, which integrates a new PI-FOSM sliding surface and a continuous control law, is developed. The crucial parameters of the controller are optimally selected by Bat algorithm so that the nearly optimal performance of the controller can be achieved. In addition, the unknown system dynamics is approximated by using a radial basic function neural network (RBFNN) so that the proposed controller does not require an exact model of the system. Compared with other existing sliding mode controllers for fault tolerant control system, the proposed method provides very strong robustness, low oscillation, fast convergence and high precision. The superior performance of the proposed robust fault tolerant controller is proved through simulation results for attitude control of a spacecraft.

Keywords: PI controller; Full-order sliding mode; Fault diagnosis; Fault tolerant control; Nonlinear Systems; Bat algorithm; Neural network.

¹*Email:* m.van@qub.ac.uk

1. Introduction

Fast development of science and technology in recent decades helps to improve quantity and quality of the products significantly. However, this increases the complexity of the operational system. Consequently, since the number of component and complexity are increased, faults occur more frequently within the system. The effects of faults not only reduce quality of products but also harm to the users and workplaces. Therefore, it is significant to investigate fault diagnosis (FD) and fault tolerant control (FTC) to improve quality of the products and reliability and safety of the system [1, 2, 3]. The integration of the FD into the system will help to detect, isolate and identify the magnitude of the faults, whilst FTC helps to reduce the effects of the faults, and thus improve performance of the system [4, 5]. In general, faults in the system can be compensated by either active or passive way [6]. In the active fault tolerant control (AFTC), faults are compensated by reconfiguring the nominal controller based on the information feedback from a FD observer [7, 8, 9, 10, 11]. This approach, however, requires an additional design of fault diagnosis/fault detection and isolation (FD/FDI) scheme, which increases the complexity and computational burden of the system. In addition, since the system can compensate the fault only until it received the fault information from the FD observer, it delays the fault compensation time and therefore, the system could become unstable during the transient period from normal operation to fault operation. In the passive fault tolerant control (PFTC), a robust controller is designed to compensate the faults, which are considered as an additional disturbance in the system, without requiring information feedback from a fault diagnosis observer [12, 13]. Compared to the AFTC, the PFTC obviously compensates the fault much faster, and thus provides higher capability to stabilize the system [13]. However, since the faults effects imposed on the nominal controller of the PFTC are heavier than that of the AFTC, the nominal controller of the PFTC requires stronger robustness against the effects of faults. Several robust controllers, which can be applied for the design of FTC, have been developed in the literature; for example PID controller [14], fuzzy logic controller [15, 16], neural network controller [17] or sliding mode controller (SMC) [18, 19, 20]. Among them, SMC is widely applied due to its robustness against

the uncertainties and disturbances [21]. However, the conventional SMC has several drawbacks that reduce its effectiveness for real applications, i.e., it provides singularity and chattering and it does not provide finite time convergence. To overcome these drawbacks, several approaches have been proposed in the literature. First, to obtain a defined finite time convergence, terminal sliding mode control (TSMC), which uses a nonlinear sliding surface instead of a linear sliding surface of the conventional SMC, has been proposed [22, 23, 24]. However, the TSMC has not considered the singularity problem. In order to tackle this issue, nonsingular terminal sliding mode control (NTSMC) has been developed [25, 26, 27]. The NTSMC has a similar property to the TSMC, but the singularity problem is removed. However, both the TSMC and NTSMC have not considered the chattering problem in their designs. In the literature, several individual approaches have been proposed to eliminate the chattering, including boundary layer method [28], disturbance observer [29, 30], or high-order sliding mode (HOSM) control [31, 32]. Although the drawbacks of the conventional SMC have been solved individually by the above-mentioned approaches, no approaches in the literature could solve all the drawbacks simultaneously. Recently, full-order sliding mode control(FOSMC), which employs the full order of the state variables into the design of the sliding surface, has been developed to resolve this issue [33]. The FOSMC takes all the weaknesses of the conventional SMC into account and solves them all simultaneously. Generally, the conventional SMC and its advanced techniques, i.e., the NTSMC or FOSM, provide high robustness, but, unfortunately, they provide high steady-state errors. In recent years, by taking the merits of the integral component as in the design of the PI [34] or PID [35, 36] controllers, some approaches have demonstrated that the robustness of the SMC can be improved and the steady-state errors can be reduced by adding an integral term of the sliding surface into the design of the SMC [37, 38, 39]. This statement has been proved in the previous work [39]. However, the work in [39] only considers the linear sliding surface in the design, and thus the aforementioned drawbacks of the conventional SMC still exist. On the other hand, one of the most challenges in the design of robust controller is to obtain an exact dynamic model of the system, which is usually difficult to obtain in advance in real applications [14, 15, 16, 17, 18, 19, 20]. In an attempt to approximate the unknown system model,

universal approximation techniques based on neural networks (NNs) or fuzzy logic have been utilized [40, 41, 42, 43, 44]. In this paper, a PFTC based on a novel PI-
full order sliding mode (PI-FOSM) control is proposed to integrate the merits of the
65 full-order sliding surface and the PI sliding surface. However, similar to the properties
of the PI controller, the performance of the PI-FOSMC depends hugely on the selection
of the proportional and integral gains of the PI sliding surface. In the literature,
several approaches based on fuzzy logic [45], genetic algorithm [46], particle swarm
optimization (PSO) [47], or bee algorithm [48] have been developed to select the op-
70 timal parameters of the conventional PI and PID controllers. In recent few years, a
relatively new swarm intelligent technique, called Bat algorithm (BA), has been intro-
duced [49]. The BA computing employs a frequency tuning method to increase the
searching diversity of the solutions. It has been successfully utilized as a new power-
ful heuristic optimization method to solve many optimization problems in the practical
75 applications [50, 51]. The superior performance of the BA compared to other heuristic
optimization methods has been demonstrated in [52]. Therefore, the BA is employed
in this paper to effectively select the gains of the proposed sliding surface to enhance
the performance of the system. In addition, an adaptive radial basis function neural
network (RBFNN) is employed to exclude the requirement of the exact model of the
80 system and the prior knowledge of the fault in formation in the design of the proposed
controller. In summary, the major contributions of this paper are as follows:

- APFTC based on a novel PI-FOSM controller, which combines a PI-FOSM slid-
ing surface and a continuous control law, is proposed. This design mechanism
provides many advantages compared to the existing approaches. For example,
85 compared to the conventional SMC [18, 19, 20], the proposed technique over-
comes all the existing drawbacks of the conventional SMC. Compared to the
NFTSMC [25, 26, 27], or FOSMC [33], the proposed method provides lower
steady-state error and faster convergence. Compared to the HOSMC [31, 32],
the proposed method has the same property in terms of chattering reduction, but
90 provides higher robustness against the effects of the uncertainties and faults.
- The crucial parameters of the proposed PI-FOSM controller, i.e., the proportional

and integral gains, are optimally selected off-line using BA so that the nearly optimal performance of the system can be achieved. This results in higher tracking performance of the proposed approach compared to the conventional PID-SMC [39].

95

- The unknown components in the system dynamics are approximated using an adaptive RBFNN so that the proposed controller does not require an exact model of the dynamic system. Since the exact dynamic model of the system is very difficult to obtain in advance in the practical systems, the use of RBFNN in this paper is useful because it makes the proposed approach more applicable for wide practical applications.

100

The remainder of this paper is organized as follows. Section 2 introduces some preliminaries. Problem statement is described in section 3. Section 4 presents the design of the finite-time integral sliding mode control. The nominal controller of the integral sliding mode control based on a finite-time backstepping control is presented in section 5. The effectiveness of the proposed approach is demonstrated in section 6. Section 7 provides conclusions and proposes future works.

105

2. Problem statement

Without loss of generality, the following second-order dynamical model is considered:

$$\begin{aligned} X_1 &= X_2 \\ X_2 &= f(X) + G(X)u + \Delta \end{aligned} \tag{1}$$

110

where $X_1 = (x_1, \dots, x_n)^T \in \mathfrak{R}^n$, $X_2 = (x_{n+1}, \dots, x_{2n})^T \in \mathfrak{R}^n$ and $X = (X_1, X_2)^T$ are the state vectors, $f(X) \in \mathfrak{R}^n$ and $G(X) \in \mathfrak{R}^{n \times m}$ are the dynamic nonlinear smooth function with $f(0) = 0$. $u = (u_1, \dots, u_m) \in \mathfrak{R}^m$ with $m \geq n$ is the actuator inputs, $\Delta = (\Delta_1, \dots, \Delta_n)^T \in \mathfrak{R}^n$ represents the system uncertainties and/or disturbances, and $(\cdot)^T$ denotes the transpose of a vector or a matrix.

115

Remark 1. *The dynamic model under consideration (1) introduces a general second-order mechanical system. Therefore, the controller developed for this model can be*

applicable for several practical systems such as surface vessels, robot manipulators, mobile robot and aircraft, etc.

In this paper, we consider actuator faults because they are the most popular faults and have severe effects on the system [26]. When a fault occurs, the actual control input of the system u differs from the designed input u_d . The relationship between them can be described as

$$u = u_d + \rho(t - t_f)\delta u(X, t) \quad (2)$$

where $\rho(t - t_f)\delta u(X, t)$ represents the unexpected actuator fault components, and t_f denotes the time instant that the fault occurs. The function $\rho(t - t_f)$ characterizes the time profile of the fault, and $\delta u(X, t)$ is bounded but uncontrollable portion of the actuator output. Two types of faults have been widely considered in the literature:

(i) Abrupt Faults:

$$\rho(t - t_f) = \begin{cases} 0, & t \leq t_f \\ 1, & t > t_f \end{cases} \quad (3)$$

(ii) Incipient Faults:

$$\rho(t - t_f) = \begin{cases} 0, & t \leq t_f \\ 1 - e^{-\beta(t - t_f)}, & t > t_f \end{cases} \quad (4)$$

where $\beta > 0$ represents the unknown fault evolution rate. A small value of β represents the incipient fault, while a large value of β characterizes the abrupt faults.

Let $u_d = (u_{d1}, \dots, u_{dm}) \in \mathfrak{R}^m$ be the designed actuator input, and $\delta_u = (\delta_{u1}, \dots, \delta_{um}) \in \mathfrak{R}^m$ the actuator faults. When a fault exists in the system, (1) can be described as:

$$\begin{aligned} X_1 &= X_2 \\ X_2 &= f(X) + G(X)(u_d + \delta u) + \Delta \end{aligned} \quad (5)$$

The system (5) can be rewritten as [3]:

$$\begin{aligned} X_1 &= X_2 \\ X_2 &= f(X) + G(X)u_d + M(X, \delta u) + \Delta \end{aligned} \quad (6)$$

where $M(X, \delta u) \in \mathfrak{R}^n$ is utilized to represent the effects of the faults in the system. The
 125 aim of this paper is to design a robust FTC law $u = u_d$ such that the desired performance
 of the system can be achieved for both normal and fault operations.

3. PFTC based on PI-Full Order Sliding Mode Control

In the design of the PFTC, the effects of faults in the system are usually treated as
 the effects of an additional disturbance, and a robust PI-FOSM is developed to com-
 130 pensate all the effects of the uncertainties, disturbances and faults in the system. As
 a similar design procedure to the conventional SMC [24], the design of the PI-FOSM
 controller consists of two steps. In the first step, a novel PI-FOSM sliding surface is se-
 lected. Then, a continuous reaching law is reconstructed based on the selected sliding
 surface to get a continuous system.

For the first step, a PI-FOSM sliding surface is selected as:

$$s_{PI-FOSM} = K_p s_{FOSM}(t) + K_i \int_0^t s_{FOSM}(t) dt \quad (7)$$

where, the full-order sliding surface is selected as [33]:

$$s_{FOSM} = \dot{X}_2 + c_2 |X_2|^{\alpha_2} \text{sign}(X_2) + c_1 |X_1|^{\alpha_1} \text{sign}(X_1) \quad (8)$$

135 where $s_{PI-FOSM}$ is the sliding variable, c_1, c_2, α_1 and α_2 are constants, which are se-
 lected as in [33]. K_p and K_i are the proportional and integral gains. The main idea of
 the proposed PI-FOSM sliding surface in (7) compared to the conventional FOSMC is
 to integrate an integral component to improve the robustness and transient response,
 while reducing the steady-state errors of the system. The following assumptions are
 140 made for the design of the PI-FOSM controller.

Assumption 1. *The system uncertainties and faults and their derivative are bounded
 by:*

$$|M(X, \delta u) + \Delta| \leq \Xi \quad (9)$$

$$\left| \frac{d}{dt} (M(X, \delta u) + \Delta) \right| \leq \Pi \quad (10)$$

where Ξ and Π are constants.

The above assumption is a generic assumption and has been widely utilized in the design of FTC system [53]. This condition states that the FTC is designed for the situations where the system is not exploding. It means the Lipschitz condition is practically satisfied in the considered operational region [53]. This assumption may not be satisfied for some conditions when the changing rates of the faults are high. In these conditions, the system might become unstable. Fortunately, due to the benefits of the proposed PI-FOSM sliding surface, the system responses against the effects of the faults very quick, and thus the stability of the system can be recovered quickly. This is one of the major contributions of the paper. This exciting feature will be verified in the results and discussions section.

To obtain the desired performance, the following controller is proposed for the system (6):

$$u_{PI-FOSM} = -\frac{1}{K_p} G^+(X)(u_{eq} + K_p u_r) \quad (11)$$

where,

$$u_{eq} = K_p f(X) + K_p (c_2 |X_2|^{\alpha_2} \text{sign}(X_2) + c_1 |X_1|^{\alpha_1} \text{sign}(X_1)) + K_i \int_0^t (\dot{X}_2 + c_2 |X_2|^{\alpha_2} \text{sign}(X_2) + c_1 |X_1|^{\alpha_1} \text{sign}(X_1)) dt \quad (12)$$

and,

$$\dot{u}_r = (\Pi + \zeta) \text{sign}(s_{PI-FOSM}) \quad (13)$$

where the pseudoinverse $G^+(X) = G^T(X)[G(X)G^T(X)]^{-1}$, the initial value of the reaching phase $u_r(0) = 0$. Π is defined as in (10) and ζ is a positive constant. The stability and convergence of the system is stated in Theorem 1.

Theorem 1. Consider the nonlinear dynamic model (6). If the proposed control laws in (11)-(13) are employed for the system (6), then the stability of the system and the convergence of the tracking errors are guaranteed.

Proof. Substituting the composite control law (11)-(13) into the sliding surface (7),

one yields

$$\begin{aligned}
s_{PI-FOSM} &= K_p(\dot{X}_2 + c_2|X_2|^{\alpha_2}\text{sign}(X_2) + c_1|X_1|^{\alpha_1}\text{sign}(X_1)) + K_i \int_0^t s_{FOSM} dt \\
&= K_p(f(X) + G(X)u_{PI-FOSM} + M(X, \delta u) + \Delta + c_2|X_2|^{\alpha_2}\text{sign}(X_2) + c_1|X_1|^{\alpha_1}\text{sign}(X_1)) \\
&\quad + K_i \int_0^t s_{FOSM} dt \\
&= K_p(M(X, \delta u) + \Delta - u_r)
\end{aligned} \tag{14}$$

Differentiating (14) with respect to time and combining the result with (13), we have

$$\dot{s}_{PI-FOSM} = K_p \left(\frac{d}{dt} (M(X, \delta u) + \Delta - u_r) - (\Pi + \zeta) \text{sign}(s_{PI-FOSM}) \right) \tag{15}$$

Consider the following Lyapunov function candidate:

$$V = \frac{1}{2K_p} s_{PI-FOSM}^T s_{PI-FOSM} \tag{16}$$

Differentiating the Lyapunov function (16) with respect to time and combining the result with (15), we have

$$\begin{aligned}
\dot{V} &= s_{PI-FOSM}^T \left(\frac{d}{dt} (M(X, \delta u) + \Delta - u_r) - (\Pi + \zeta) \text{sign}(s_{PI-FOSM}) \right) \\
&= \frac{d}{dt} (M(X, \delta u) + \Delta - u_r) s_{PI-FOSM} - (\Pi + \zeta) |s_{PI-FOSM}| \\
&= \left(\frac{d}{dt} (M(X, \delta u) + \Delta - u_r) s_{PI-FOSM} - \Pi |s_{PI-FOSM}| \right) - \zeta |s_{PI-FOSM}| \\
&< -\zeta |s_{PI-FOSM}| < 0
\end{aligned} \tag{17}$$

Therefore, based on the Lyapunov criterion, we can verify that the stability and convergence of the sliding surface, i.e., $s_{PI-FOSM}$, is guaranteed under the composite control law (11-13) despite the existing of the uncertainties, disturbances and faults. This completes the proof.

Once the ideal sliding mode $s_{PI-FOSM} = 0$ is established, then based on (7), we have $s_{FOSM} = 0$. As a result, from (7), we have

$$\begin{aligned}
\dot{X}_1 &= X_2 \\
\dot{X}_2 &= -c_2|X_2|^{\alpha_2}\text{sign}(X_2) - c_1|X_1|^{\alpha_1}\text{sign}(X_1)
\end{aligned} \tag{18}$$

With suitable chosen parameters of c_1, c_2, α_1 and α_2 , the tracking errors of the system (18) can converge to its equilibrium point $X = [X_1, X_2]^T = [0, 0]^T$ from any initial condition $X(0) \neq 0$ along the terminal sliding surface $s_{FOSM} = 0$ in a finite-time [33].

4. Adaptive Neural PI-Full Order Sliding Mode Control

4.1. Radial Basis function Neural Network

The RBFNN is a three-layer forward neural network, consisting of one input layer, one hidden layer, and one output layer. The output of the RBFNN can be defined as:

$$f(\tau) = \theta^T \psi(\tau) + \varepsilon(\tau) \quad (19)$$

where τ and $f(\tau)$ are the input and output of the RBFNN, respectively. $\theta^T \in \mathfrak{R}^{n \times l}$ is the weight matrix of the hidden nodes. $\psi(\tau) \in \mathfrak{R}^l$ is the nonlinear function of the hidden nodes. $\varepsilon(\tau)$ is an approximating error with bound. A Gaussian function is chosen for the nonlinear function

$$\psi_i(\tau) = \exp\left(\frac{-(\tau - \mu_i)^T(\tau - \mu_i)}{\sigma_i^2}\right), i = 1, 2, \dots, l \quad (20)$$

where μ and σ are the center and width of the Gaussian function, respectively. The representative schematic of a RBFNN is illustrated in Fig. 1. It is noted that the approximation capability of the RBFNN depends on the the selection of the number of hidden nodes in the hidden layer. The higher number of the nodes, the better approximation achieves, but higher computational burden needs, and vice versa. In this paper, the selection of nodes are based on experience to guarantee both acceptable approximation capability and computational burden.

4.2. Design of PFTC based on Adaptive Neural PI-Full Order Sliding Mode Control

In the control laws (11)-(13), the design is based on the assumption that the system dynamics $f(X)$ is known and the bounded value Π in the Assumption 1 can be obtained in advance. However, these parameters are difficult to obtain in advance in the practical engineering applications. In this paper, a RBFNN is employed to approximate these parameters.

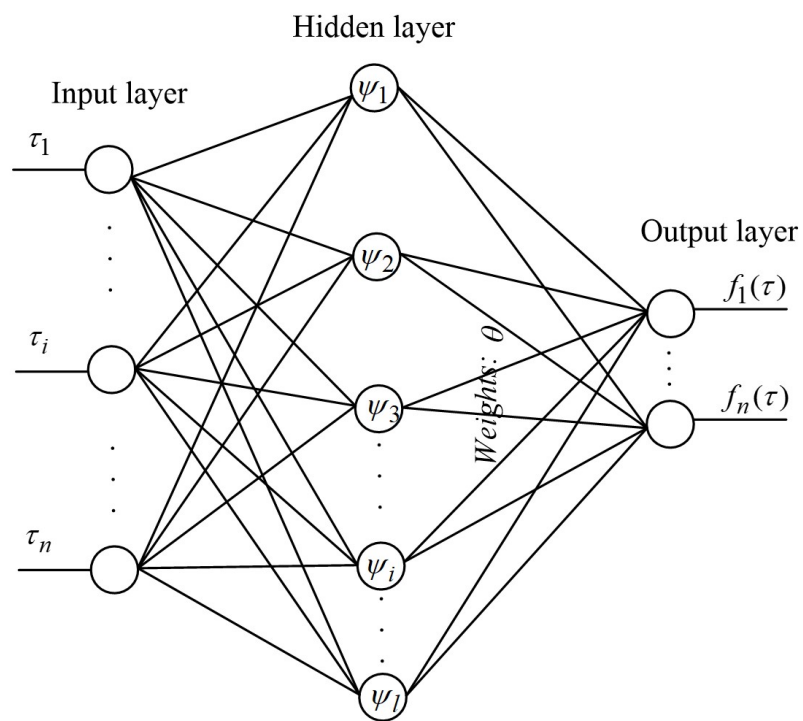


Figure (1) Representative schematic of radial basis function neural network

The unknown components in the system (6) can be represented as

$$\Omega(X) = f(X) + M(X, \delta u) + \Delta \quad (21)$$

Denote $\hat{\Omega}(X)$ as the estimation function of the $\Omega(X)$. According to [54], the estimation, $\hat{\Omega}(X)$, can be represented by an integral neural network:

$$\hat{\Omega}(X) = \int_0^t \hat{\theta}^T \psi(X) dt \quad (22)$$

where $\hat{\theta}$ denotes the adjustable parameter vector.

Then, the optimal parameter θ^* can be defined as

$$\theta_f^* = \operatorname{argmin} \left\{ \sup_{x \in U_x} |\Omega(X) - \hat{\Omega}(X, \hat{\theta})| \right\} \quad (23)$$

Consequently, $\Omega(X)$ is approximated to arbitrary accuracy by the RBFNN (22) as stated in Lemma 1.

Lemma 1. [43, 44] *For any given real continuous function $\Omega(X)$ on a compact set $U_x \in \mathfrak{R}^n$ and an arbitrary $\varepsilon > 0$, there exists a neural approximator $\hat{\Omega}$ in the form of (22) such that*

$$\sup_{x \in U_x} |\Omega(X) - \hat{\Omega}(X, \hat{\theta})| < \varepsilon \quad (24)$$

Due to the approximation capability of the RBFNN, system (6) can be rewritten as:

$$\begin{aligned} \dot{X}_1 &= X_2 \\ \dot{X}_2 &= \int_0^t \theta(t)^*{}^T \psi(X) dt + G(X)u_d + \omega \end{aligned} \quad (25)$$

where $\omega = f(X) + M(X, \delta u) + \Delta - \int_0^t \theta(t)^*{}^T \psi(X) dt$ denotes the lumped uncertainty. It is assumed that the lumped uncertainty and its derivative are to be bounded by unknown constants, i.e., $\omega \leq \bar{\omega}$ and $\frac{d}{dt}(\omega) \leq K_a$. Due to the approximating capability of the neural network, this assumption is usually satisfactory in the practical systems [43, 44].

An adaptive neural PI-FOSM control law can be designed as

$$u_{aPI-FOSM} = -\frac{1}{K_p} G^+(X)(u_{aeq} + K_p u_{ar}) \quad (26)$$

where,

$$u_{aeq} = K_p \int_0^T \hat{\theta}^T(t) \psi(X) dt + K_p (c_2 |X_2|^{\alpha_2} \text{sign}(X_2) + c_1 |X_1|^{\alpha_1} \text{sign}(X_1)) \\ + K_i \int_0^t (\dot{X}_2 + c_2 |X_2|^{\alpha_2} \text{sign}(X_2) + c_1 |X_1|^{\alpha_1} \text{sign}(X_1)) dt \quad (27)$$

and,

$$\dot{u}_{ar} = (\hat{K}_a + \zeta) \text{sign}(s_{PI-FOSM}) \quad (28)$$

where the adaptive laws are designed as

$$\dot{\hat{K}}_a = \frac{1}{\lambda} |s_{PI-FOSM}| \quad (29)$$

$$\dot{\hat{\theta}} = \frac{1}{\alpha} s_{PI-FOSM} \psi(X) \quad (30)$$

where λ and α are the constant adaptive rates.

Theorem 2. Consider the dynamic system (6) and the selected sliding surface (7). If the control laws in (26-28) with the tuning laws in (29)-(30) are used to control the system (6), then the stability of the system and the convergence of the tracking errors are guaranteed.

Proof.

From (7) and (26), (27), the sliding surface (7) becomes

$$s_{PI-FOSM} = K_p \left(\int_0^t \tilde{\theta}^T(t) \psi(X) dt + \omega + u_{ar} \right) \quad (31)$$

where $\tilde{\theta} = \theta^* - \hat{\theta}$ are the approximation errors of the NN's weights.

Differentiating the sliding surface (31) with respect to time, we have

$$\dot{s}_{PI-FOSM} = K_p \left(\tilde{\theta}^T(t) \psi(X) + \frac{d}{dt}(\omega) + \dot{u}_{ar} \right) \quad (32)$$

Consider the following Lyapunov function candidate

$$V = \frac{1}{2K_p} s_{PI-FOSM}^T s_{PI-FOSM} + \frac{1}{2} \lambda \tilde{K}_a^T \tilde{K}_a + \frac{1}{2} \alpha \tilde{\theta}^T \tilde{\theta} \quad (33)$$

195 where $\tilde{K}_a = \hat{K}_a - K_a$ is the adaptation gain error.

Differentiating the Lyapunov function (33) with respect to time and combining the results with (31), we have:

$$\begin{aligned}\dot{V} &= \frac{1}{K_p} s_{PI-FOSM}^T \dot{s}_{PI-FOSM} + \lambda \tilde{K}_a^T \dot{\tilde{K}}_a + \alpha \tilde{\theta}^T \dot{\tilde{\theta}} \\ &= s_{PI-FOSM}^T \left(\tilde{\theta}^T(t) \psi(X) + \frac{d}{dt}(\omega) + \dot{u}_{ar} \right) + \lambda (\hat{K}_a - K_a) \dot{\hat{K}}_a + \alpha \tilde{\theta}^T \dot{\tilde{\theta}}\end{aligned}\quad (34)$$

Inserting (28)-(30) into (34), we have

$$\begin{aligned}\dot{V} &= \frac{1}{K_p} s_{PI-FOSM}^T \dot{s}_{PI-FOSM} + \lambda (\hat{K}_a - K_a) \dot{\hat{K}}_a - \alpha \tilde{\theta}^T \dot{\tilde{\theta}} \\ &= s_{PI-FOSM} \frac{d}{dt}(\omega) - K_a |s_{PI-FOSM}| - \zeta |s_{PI-FOSM}| \\ &< -\zeta |s_{PI-FOSM}| < 0\end{aligned}\quad (35)$$

Therefore, based on the Lyapunov criterion, we can verify that the stability of the tracking errors is guaranteed under the control law (26-28) despite the existing of the uncertainties and faults. This completes the proof.

Remark 2. *Although the proposed PI-FOSMC provides several advantages, its design based on the Assumption 1, which is difficult to obtain in the practical applications, prevents its applicability for wide applications. In order to overcome this limitation, an adaptive neural network is used in this paper. Although the use of NN is very popular in the literature to approximate the unknown function, however, it is still important to represent it here since it can help to facilitate the implementation of the proposed approach. In addition, the design of the integral NN in this paper is considered as a novelty because, in this paper, the integral NN is used instead of the conventional NN, as described in (27).*

Remark 3. *The design of the proposed PI-FOSMC requires the measurement of position, velocity and acceleration of the state variables. In the case that the velocity and acceleration cannot be measurable, the second-order exact differentiator method [32] can be applied to obtain the estimations of these parameters.*

Remark 4. *Some approaches in the literature used learning techniques, i.e., neural network or fuzzy logic, to compensate for the unknown function in the system with-*

out using sliding mode control [55, 56]. These approaches might provide good per-
 215 *formance for compensating the uncertainty and disturbance, but might not provide*
good performance for tackling the effects of faults since the effects of faults are much
stronger. This paper integrates the SMC technique with the RBFNN to get the merits
of both. This is particularly important for FTC system since the SMC possesses high
robustness against the effects of faults.

220 **5. Design of Optimal Adaptive PI-Full Order Sliding Mode Control**

Similar to the analysis of the PI controller [34], the selection of the proportional
 and integral gains, i.e., K_P and K_I , of the PI sliding surface affects on the performance
 of the whole system significantly. Therefore, in order to get the best performance, these
 parameters need to be optimally selected. In this paper, we employ Bat algorithm (BA)
 225 to select these parameters automatically.

BA [49] is a heuristic method based on the echolocation behavior of bats. In BA,
 the initial position χ_i , velocity v_i and frequency f_i are initialized for each bat b_i . For
 each time step t , the movement of the virtual bats is given by updating their velocity
 and position using the following equations:

$$f_i = f_{min} + (f_{min} - f_{max})\phi \quad (36)$$

$$v_i^j(t) = v_i^j(t-1) + [\chi_i^j(t-1) - best]f_i \quad (37)$$

$$\chi_i^j(t) = \chi_i^j(t-1) + v_i^j(t) \quad (38)$$

where f_i and $v_i^j(t)$ are the frequency and velocity of variable j for bat i at time step
 t . The result of f_i is used to control the pace and range of the movement of the bats.
 $\chi_i^j(t-1)$ is the position vector of variable j for bat i at time step $t-1$, and j denotes
 a randomly generated number within the interval $[0; 1]$. The variable $best$ represents
 230 the current global best location (solution) for decision variable j , which is achieved by
 comparing all the solutions provided by the M bats.

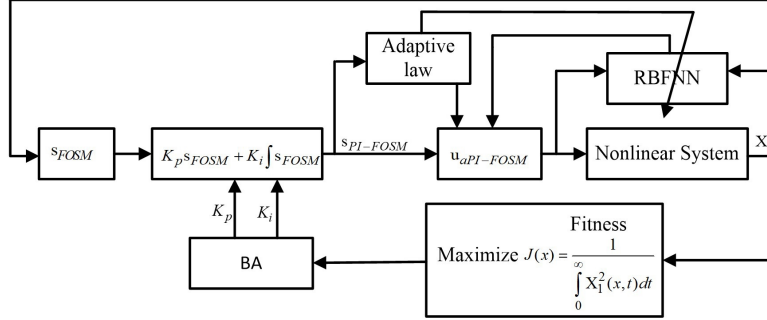


Figure (2) Structure of the proposed optimal adaptive neural PI-full order sliding mode controller

In order to improve local search capability of the algorithm, once a solution is selected among the current best solutions, a new solution for each bat is generated locally using a random walk

$$\chi_{new} = \chi_{old} + \rho A^t \quad (39)$$

where $\rho \in [-1, 1]$ is a random number, while A^t is the average loudness of all bats at this time step. The loudness A and pulse emission rate r are updated as a bat gets closer to its target. Loudness A is decreased while pulse emission rate r is increased as the following two equations:

$$A^{t+1} = \vartheta A^t \quad (40)$$

$$r^{t+1} = r_i^0 (1 - e^{-\chi t}) \quad (41)$$

where χ and ϑ are constants, r_i^0 is the initial pulse emission rate value of the i th bat. The basic steps of the selection of the parameters K_p and K_i are described as in Appendix A. The structure of the proposed optimal adaptive neural PI-FOSM control is depicted in Fig.2.

235

Remark 5. Other heuristic optimization methods such as genetic algorithm (GA), particle swarm optimization (PSO), etc., can be used to get optimal parameters as a similar way as BA. However, this paper employs BA because it provides faster convergence and

240 *closely optimal value. Due to the inherent property of a heuristic optimization method, the use of BA to optimize parameters may not give a global optimal performance for the system. However, this approach at least provides a local optimal value so that the performance of the system is usually better than the system with the parameters chosen randomly or chosen based on experiences. Since the optimization process is the time consuming process, it is done offline to get the nearly optimal parameters. The*
 245 *obtained parameters are then used for online process.*

6. Results and Discussions

While the proposed method can be applied for a class of second-order nonlinear system such as spacecraft, robot manipulator, its performance is tested for a spacecraft as a case study. The considered nonlinear dynamic model of a spacecraft used in this
 250 paper is taken as the same model as in Liang Y et al [57]. The dynamic model of a spacecraft can be described as the same form as the system (1)-(2) with $n = 3$ and the system dynamic functions are [57]: $f(X) = [f_1(X), f_2(X), f_3(X)]^T$, where $X_1 = (x_1, x_2, x_3)^T = (\phi_a, \theta_a, \psi_a)^T$, $X_2 = (x_4, x_5, x_6)^T = (\dot{\phi}_a, \dot{\theta}_a, \dot{\psi}_a)$, and $X = (X_1, X_2)^T = (u_1, u_2, u_3, u_4)^T$, and $\Delta = [\Delta_1, \Delta_2, \Delta_3]^T$. The three parameters ϕ_a, θ_a , and ψ_a indicate
 255 the three Eulers angles with respect to x, y, z axes of the spacecraft, respectively. u denotes actuators, and the components $f(X)$ and $G(X)$ are defined as in Appendix B. The initial condition is given as $X(0) = [-0.7, -0.07, 1.5, 0.3, 1.3, -0.2]^T$ and the disturbance is generated as $\Delta = 0.5[\sin(t), \cos(2t), \sin(3t)]^T$.

In this paper, the performance of the system is simulated in three different work-
 260 ing scenarios. First, from the starting time (0s) to 10s, we assume that the system works in normal condition. Second, from the time 10s to 30s, we assume that a bias fault $\Delta u_1 = 5 + 15\cos(t)$ occurs in the first actuator. Third, from the time 20s to 30s, the partial lost $\Delta u_2 = -0.8u_2$ is assumed in the second actuator (the second actuator lost 80% of its effectiveness). It means that we assumed an actuator fault
 265 $\Delta u = [5 + 15\cos(t)(10s), -0 : 8u_2(20s), 0, 0]^T$ occurs in the system. In this simulation, only abrupt faults are simulated since the effects of the abrupt faults are much heavier than that of the incipient faults. In addition, in order to verify the superior

performances of the proposed PI-FOSM controller and optimal PI-FOSM controller, we compare these methods with the recent advanced sliding mode control techniques such as the nonsingular fast terminal sliding mode control (NFTSMC) [25, 26] and the FOSMC [33]. The design of the NFTSMC and FOSMC are represented in Appendix C and Appendix D, respectively. The selected parameters of the NFTSMC and the FOSMC are shown in Table 1. The parameters of the NFTSMC were chosen as the same values as the ones used in [25, 26], while the parameters of the FOSMC were chosen as the same values as the ones used in [33]. For fairer in comparison, the parameters of the PI-FOSMC are selected as the same as the parameters of the FOSMC with $K_p = 20$ and $K_i = 5$, as shown in Table 1. The values of the K_p and K_i are selected based on trial and error experiments. The parameters of the Optimal PI-FOSMC are selected as the same as the PI-FOSMC, but the parameters of K_p and K_i are automatically selected based on the BA. The parameters of the BA are selected as in Table 1. Under the effects of the assumed faults, the system performance under the inputs of the four controllers, i.e., the NFTSMC, FOSMC, PI-FOSM and the optimal PI-FOSM, are shown in Figs. 3, 4, 5 and 6. Particularly, the position tracking performances of the four controllers are shown in Fig. 3, whilst Fig. 4 shows the velocity tracking performance. From Figs. 3 and 4, some important results are discussed as follows. First, the conventional NFTSMC provides good performance when the system works in normal operation. However, when a fault occurs, the system tends to be out of control; the tracking errors are very big. It means that the NFTSMC has weak robustness against the effects of faults. Second, the FOSMC provides better performance than the NFTSMC in terms of tackelling the effects of the faults. Third, the PI-FOSMC and the Optimal PI-FOSMC provide much better performance compared to the NFTSMC and the FOSMC. This results proved that the integration of the PI sliding surface into the FOSMC provides higher robustness and faster transient response against the effects of faults. This feature is very important in the design of FTC system because it helps stabilize the system quicker. Fourth, the comparison results between the PI-FOSMC and the Optimal PI-FOSMC show that the selection of the proportional and integral gains affects on the performance of the system significantly. Since the Optimal PI-FOSMC selects the proportional and derivative gains optimally, its performance is always better

than the PI-FOSMC, where the gains are selected randomly or based on experiences.
300 The convergence of the fitness function of the BA of the Optimal PI-FOSMC is shown
in Fig. 5. Figure 6 shows the control inputs of the four controllers. It can be seen from
Fig. 6 that the Optimal PI-FOSMC provides continuous control input. In summary,
from the results discussed above, two important conclusions can be made. First, the
integration of the PI and the FOSM sliding surfaces increases robustness and transient
305 response against the faults effects in the system. Second, the employment of the BA to
automatically select the parameters help to improve the performance of the system.

Remark 6. *There is no particular reason to assume the values of disturbance and
fault components, i.e., Δ and δu , used in this paper. However, these parameters are
selected to model a scenario of uncertainty and fault effects, which can be occurred in
310 the practical applications.*

Remark 7. *In this paper, a robust FTC is designed to compensate for the effects of
actuator or component faults only, in which the actuator faults are particularly con-
cerned. The proposed method might not be able to compensate for the effects of sensor
faults. Solving the effects of sensor faults will be investigated in our future work. In the
315 literature, fault diagnosis for sensor faults has been investigated, for example in [58].*

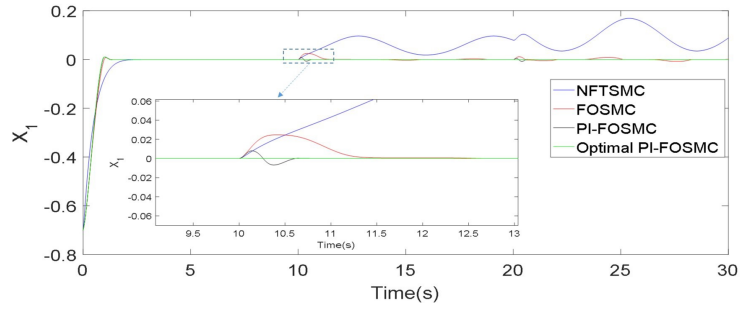
Remark 8. *From the design structure of the NFTSMC, FOSMC and the proposed PI-
FOSMC, it is obvious to see that the computational burden of the proposed PI-FOSMC
is higher than that of the NFTSMC and the FOSMC (the FOSMC is higher than the
NFTSMC). However, with the recent advance on hardware development for computing,
320 it is believed that all three controllers, i.e., NFTSMC, FOSMC and PI-FOSMC, are
implementable real time in the practical applications.*

7. Conclusions

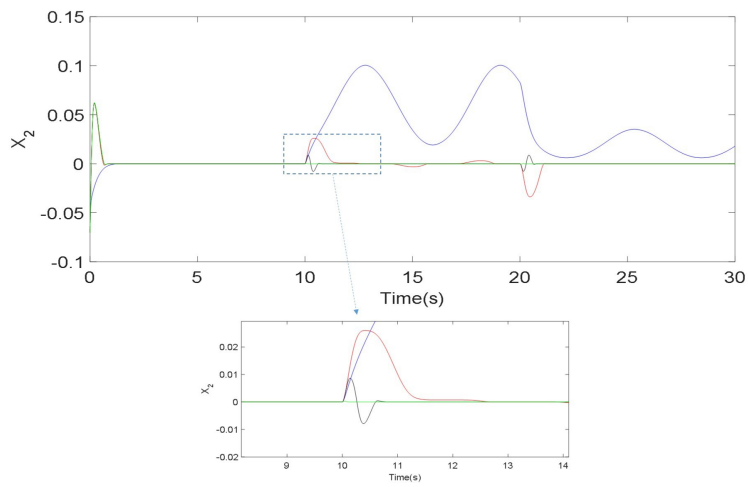
This paper presented a robust fault tolerant control based on the passive approach
for a class of second-order nonlinear system using an optimal adaptive neural PI-
325 FOSMC. The PI-FOSMC was designed to integrate the benefits of the FOSM and PI

Table (1) Selected parameters of the controllers

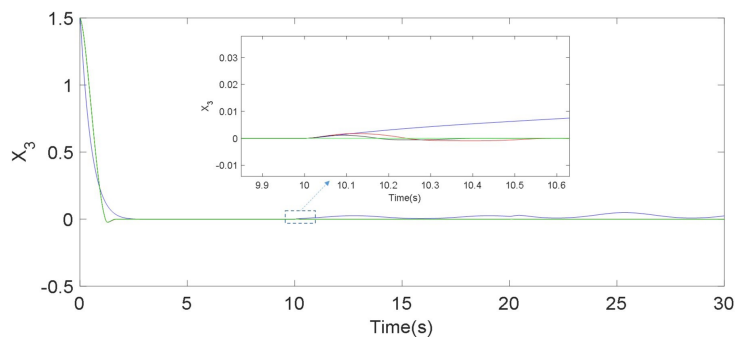
Controller	Parameters	Value
NFTSNC	σ_1	$diag\{10, 10, 10\}$
	σ_1	$diag\{5, 5, 5\}$
	$\alpha, p, q, \Pi + \zeta, \Theta$	1.4, 9, 7, 20, 1000
FOSMC	$c_1, c_2, \alpha_1, \alpha_2$	10, 9, 9/23, 9/16
	$\Pi + \zeta, \Theta$	20, 1000
PI-FOSMC	K_p, K_i	20, 5
Optimal PI-FOSMC	$c_1, c_2, \alpha_1, \alpha_2$	10, 9, 9/23, 9/16
	Number of particle of BA	10
	Number of iteration of BA	10
	$f_{min}, f_{max}, c_1, c_2$	0, 2, 2, 2
	Emission rate $r(0)$, loudness $A(0)$	0.9, 0.95



(a)

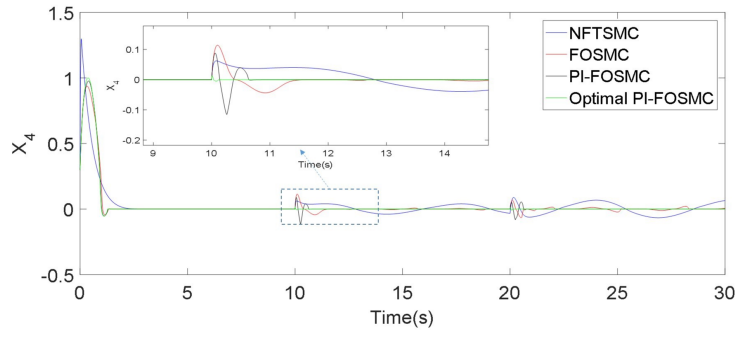


(b)

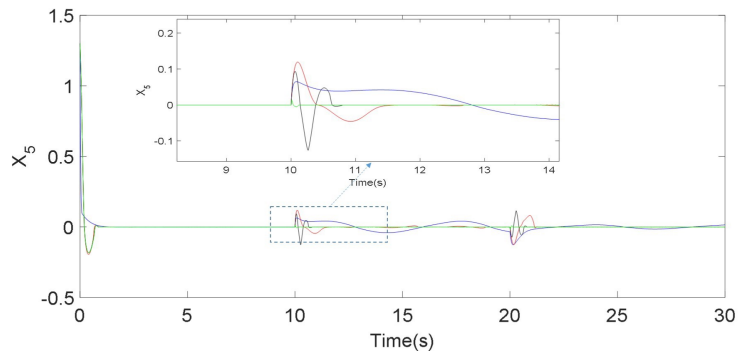


(c)

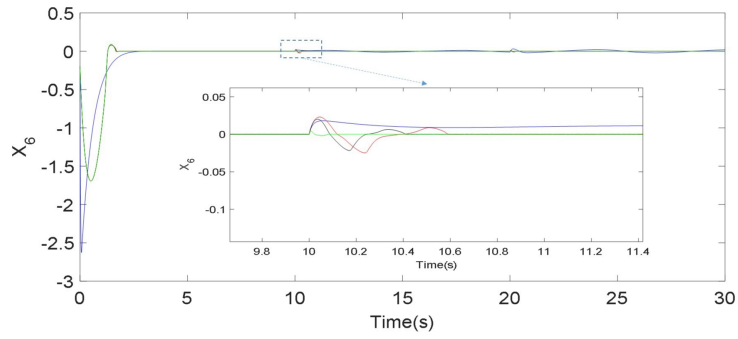
Figure (3) Time history of system states x_1, x_2, x_3 under the effect of fault δu



(a)



(b)



(c)

Figure (4) Time history of system states x_4, x_5, x_6 under the effect of fault δu

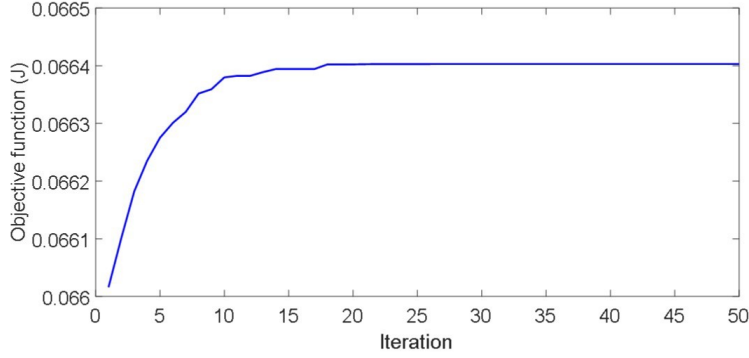


Figure (5) Time history of the BAs fitness function

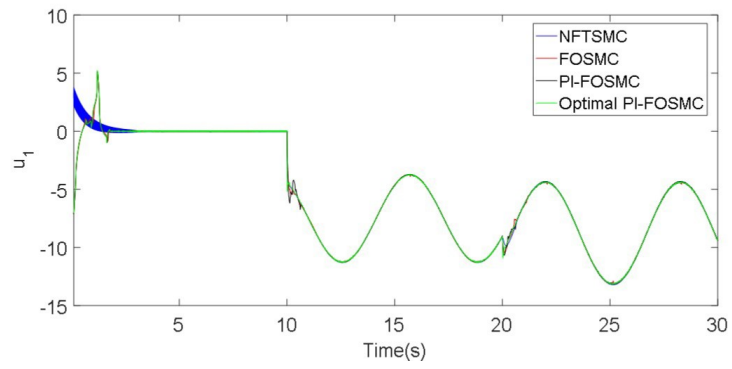
sliding surfaces. The unknown components in the system dynamics were approximated using an adaptive RBFNN so that the proposed controller can be implementable without the prior knowledge of the exact dynamic model and fault information. In addition, the crucial parameters such as the proportional and integral gains of the proposed controller were optimized using Bat algorithm so that the nearly optimal performance of the system can be achieved. While the proposed method can be utilized for a class of second-order nonlinear system, a computer simulation was performed for attitude control of a spacecraft as an example. The simulation results verified that the proposed approach, i.e., adaptive neural PI-FOSMC, provided superior fault tolerant capability compared to other advanced robust fault tolerant controllers.

Appendix A.

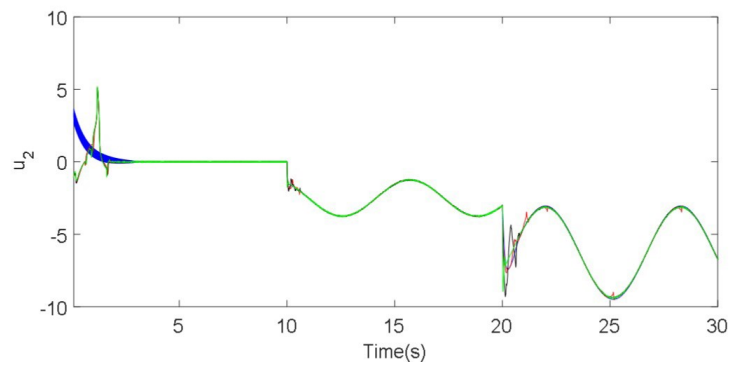
The selection of K_p and K_i based on BA are as follows:

Step 1: Sets an initial bat population P , by randomly generating a certain number of position vectors. Initializes the two dimensions of the position for the i th bat, as $x_i(t) = (x_1^i, x_2^i)$ (x_1^i represents for K_p and x_2^i represents for K_i) and the two dimensions of velocity, $v_i(t) = (v_1^i, v_2^i)$, is initialized in $[0, 1]$. Defines frequency f_i and initializes pulse emission rate r and loudness A .

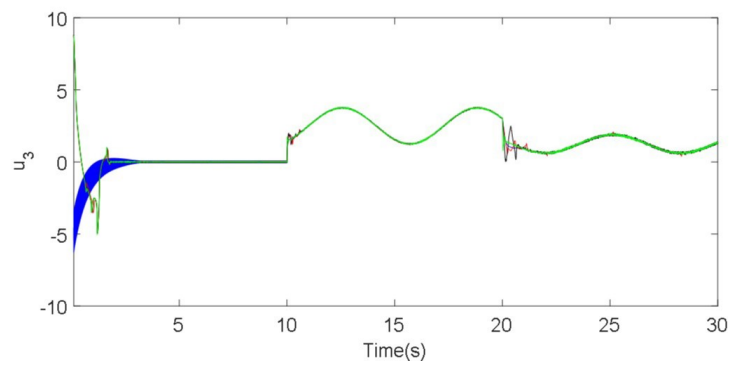
Step 2: Evaluates the fitness value of each initialized bat. In this paper, the fitness



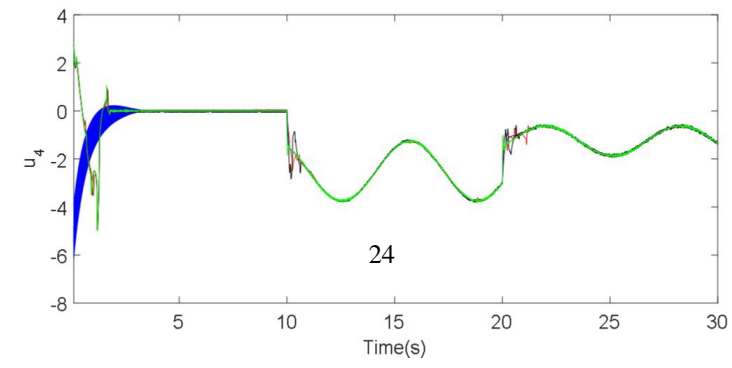
(a)



(b)



(c)



(d)

Figure (6) Time histories of four control inputs

function is calculated as

$$J(x) = \frac{1}{\int_0^{\infty} X_1^2(x,t)dt} \quad (\text{A.1})$$

Since the objective is to minimize the error X_1 , the higher the value J the better fitness will receive. After this step is finished, the best solution is stored.

345 **Step 3:** For each bat, updates bat velocity with (37) and bat position with (38) to generate a new population.

Step 4: If $\text{rand} > r_i$, then: selects a solution among best solutions and generates new local solution around the selected best solution.

Step 5: Generates new solution by flying randomly.

350 **Step 6:** If $\text{rand} < A_i$ and $J(x_i) > J(\text{best})$, then: accepts the new solution, and decreases A_i , increases r_i according to (40) and (41).

Step 7: Evaluates the fitness value of each initialized bat as a similar procedure to Step 2 and finds the best solution best .

355 **Step 8:** If the maximum number of iterations is not yet reached, increases the number of iteration and returns to Step 3. Otherwise, proceeds to Step 9.

Step 9: Selects the global best position, best , in the population as the ultimate solution. The values encoded from the global best position, best , are assigned as the optimal values for K_p and K_i .

Appendix B.

The components $f(X)$ and $G(X)$ are computed as follows [57]:

$$\begin{aligned} f_1(X) = & \omega_0 x_6 c x_3 c x_2 - \omega_0 x_5 s x_3 s x_2 + \frac{I_y - I_z}{I_x} [x_5 x_6 \\ & + \omega_0 x_5 c x_1 s x_3 s x_2 + \omega_0 x_5 c x_3 s x_1 + \omega_0 x_6 c x_3 c x_1 \\ & + \frac{1}{2} \omega_0^2 s(2x_3) c^2 x_1 s x_2 + \frac{1}{2} \omega_0^2 c^2 x_3 s(2x_1) \\ & - \omega_0 x_6 s x_3 s x_2 s x_1 - \frac{1}{2} \omega_0^2 s^2 x_2 s^2 x_3 s(2x_1) \\ & - \omega_0 s(2x_2) s x_2 s^2(x_1) - \frac{3}{2} \omega_0^2 c^2(x_2) s(2x_1)] \end{aligned} \quad (\text{B.1})$$

$$\begin{aligned}
f_2(X) &= \omega_0 x_6 s x_3 c x_1 + \omega_0 x_4 c x_3 s x_1 + \omega_0 x_6 c x_3 s x_2 s x_1 \\
&+ \omega_0 x_5 s x_3 c x_2 s x_1 + \omega_0 x_4 s x_3 s x_2 c x_1 \\
&+ \frac{I_z - I_x}{I_y} [x_4 x_6 + \omega_0 x_4 c x_1 s x_3 s x_2 + \omega_0 x_4 c x_3 s x_2 s x_1 \\
&- \omega_0 x_6 s x_3 c x_2 - \frac{1}{2} \omega_0^2 s(2x_2) s^2(x_3) c x_1 \\
&- \frac{1}{2} \omega_0^2 c x_2 s x_1 s(2x_3) + \frac{3}{2} \omega_0^2 s(2x_2) c x_1]
\end{aligned} \tag{B.2}$$

$$\begin{aligned}
f_3(X) &= \omega_0 x_4 s x_1 s x_3 s x_2 - \omega_0 x_6 c x_1 c x_3 s x_2 \\
&- \omega_0 x_5 c x_1 s x_3 c x_2 + \omega_0 x_6 s x_3 s x_1 - \omega_0 x_4 c x_3 c x_1 \\
&+ \frac{I_x - I_y}{I_z} [x_4 x_5 + \omega_0 x_4 c x_3 c x_1 - \omega_0 x_4 s x_3 s x_2 s x_1 \\
&- \omega_0 x_5 s x_3 c x_2 - \frac{1}{2} \omega_0^2 s(2x_3) c x_2 c x_1 \\
&- \frac{1}{2} \omega_0^2 s_2 x_3 s x_1 s(2x_2) - \frac{3}{2} \omega_0^2 s(2x_2) s x_1]
\end{aligned} \tag{B.3}$$

$$G(X) = \begin{bmatrix} 0.67 & 0.67 & 0.67 & 0.67 \\ 0.69 & -0.69 & -0.69 & 0.69 \\ 0.28 & 0.28 & -0.28 & -0.28 \end{bmatrix} \tag{B.4}$$

360 In the above equations, c and s denote the cos and sin functions, respectively. I_x , I_y and I_z are the moment of inertia with respect to the three body coordinate axes, ω_0 denotes the constant orbital rate.

Appendix C.

The PFTC based on NFTSMC for system (3) can be designed as follows [25, 26]:

First, the nonsingular fast terminal sliding mode surface is selected as:

$$s_{NFTSM} = X_1 + \sigma_1 X_1^{[\alpha]} + \sigma_2 X_2^{[p/q]} \tag{C.1}$$

365 where s_{NFTSM} is the sliding variable, $\sigma_1 = \text{diag}(\sigma_{11}, \sigma_{12}, \dots, \sigma_{1n}) \in \mathfrak{R}^{n \times n}$ and $\sigma_2 = \text{diag}(\sigma_{21}, \sigma_{22}, \dots, \sigma_{2n}) \in \mathfrak{R}^{n \times n}$ are positive definite matrices, p and q are positive odd

numbers and satisfying the relation $1 < p/q < 2$ and $\alpha > p/q$, $X_1^{[\alpha]} := (|X_1|^\alpha \text{sign}(X_1), \dots, |X_n|^\alpha \text{sign}(X_n)) \in \mathfrak{R}^n$, and $X_2^{[p/q]} := (|\dot{X}_1|^{p/q} \text{sign}(\dot{X}_1), \dots, |\dot{X}_n|^{p/q} \text{sign}(\dot{X}_n)) \in \mathfrak{R}^n$.

Under assumption 1, a PFTC based on NFTSMC can be designed as

$$u = -G^+(X)(u_{eq-NFTSM} + u_{re-NFTSM}) \quad (\text{C.2})$$

where,

$$u_{eq-NFTSM} = \frac{1}{\sigma_2} \frac{q}{p} |X_2|^{2-(p/q)} + \sigma_1 \gamma |X_1|^{\alpha-1} \frac{1}{\sigma_2} \frac{q}{p} |X_2|^{2-(p/q)} + f(X) \quad (\text{C.3})$$

and,

$$u_{re-NFTSM} = (\Xi + \zeta) \text{sign}(s_{NFTSM}) \quad (\text{C.4})$$

where Ξ is a constant defined in the Assumption 1. In order to alleviate the chattering phenomenon, a *sigmoid* function $\frac{e^{\Theta s_{NFTSM}} - 1}{e^{\Theta s_{NFTSM}} + 1}$ is used to replace the $\text{sign}(s_{NFTSM})$ function, where Θ is a constant.

Appendix D.

The PFTC based on FOSMC for system (5) can be designed as follows [33]:

First, the sliding surface is designed as:

$$s_{FOSM} = \dot{X}_2 + c_2 |X_2|^{\alpha_2} \text{sign}(X_2) + c_1 |X_1|^{\alpha_1} \text{sign}(X_1) \quad (\text{D.1})$$

where $s_{FOSM} \in \mathfrak{R}^n$ is the sliding variable, c_1 , c_2 , α_1 and α_2 are constants. Based on [33], the PFTC control law can be designed as follows:

$$u_{FOSM} = -G^+(X)(u_{eq-FOSM} + u_{re-FOSM}) \quad (\text{D.2})$$

where,

$$u_{eq-FOSM} = f(X) + c_2 |X_2|^{\alpha_2} \text{sign}(X_2) + c_1 |X_1|^{\alpha_1} \text{sign}(X_1) \quad (\text{D.3})$$

and,

$$\dot{u}_{re-FOSM} = -(\Pi + \zeta) \text{sign}(s_{FOSM}) \quad (\text{D.4})$$

For the stability and convergence, the interested readers can refer to [33] for more detail.

Conflict of interest

The authors declare that they have no conflict of interest.

References

- [1] J. Shao, Z. Deng, Y. Gu, Fault-tolerant control of position signals for switched reluctance motor drives, *IEEE Transactions on Industry Applications* 53 (3) (2017) 2959–2966. doi:10.1109/TIA.2017.2669190.
- [2] M. Van, H. Kang, Bearing defect classification based on individual wavelet local fisher discriminant analysis with particle swarm optimization, *IEEE Transactions on Industrial Informatics* 12 (1) (2016) 124–135. doi:10.1109/TII.2015.2500098.
- [3] M. Van, D. Wu, S. S. Ge, H. Ren, Fault diagnosis in image-based visual servoing with eye-in-hand configurations using kalman filter, *IEEE Transactions on Industrial Informatics* 12 (6) (2016) 1998–2007. doi:10.1109/TII.2016.2590338.
- [4] J. Amini, M. Moallem, A fault-diagnosis and fault-tolerant control scheme for flying capacitor multilevel inverters, *IEEE Transactions on Industrial Electronics* 64 (3) (2017) 1818–1826. doi:10.1109/TIE.2016.2624722.
- [5] Z. Gao, S. X. Ding, C. Cecati, Real-time fault diagnosis and fault-tolerant control, *IEEE Transactions on Industrial Electronics* 62 (6) (2015) 3752–3756. doi:10.1109/TIE.2015.2417511.
- [6] X. Yu, J. Jiang, Hybrid fault-tolerant flight control system design against partial actuator failures, *IEEE Transactions on Control Systems Technology* 20 (4) (2012) 871–886. doi:10.1109/TCST.2011.2159606.
- [7] Q. Fan, G. Yang, Active complementary control for affine nonlinear control systems with actuator faults, *IEEE Transactions on Cybernetics* 47 (11) (2017) 3542–3553. doi:10.1109/TCYB.2016.2569406.

- [8] G. Zhang, H. Zhang, X. Huang, J. Wang, H. Yu, R. Graaf, Active fault-tolerant control for electric vehicles with independently driven rear in-wheel motors against certain actuator faults, *IEEE Transactions on Control Systems Technology* 24 (5) (2016) 1557–1572. doi:10.1109/TCST.2015.2501354.
- 405 [9] Q. Shen, B. Jiang, V. Cocquempot, Adaptive fuzzy observer-based active fault-tolerant dynamic surface control for a class of nonlinear systems with actuator faults, *IEEE Transactions on Fuzzy Systems* 22 (2) (2014) 338–349. doi:10.1109/TFUZZ.2013.2254493.
- [10] P. Shi, M. Liu, L. Zhang, Fault-tolerant sliding-mode-observer synthesis of markovian jump systems using quantized measurements, *IEEE Transactions on Industrial Electronics* 62 (9) (2015) 5910–5918. doi:10.1109/TIE.2015.2442221.
- 410 [11] P. Y. J. Gao, Q. Shen, J. Gong, Sliding mode fault tolerant control with prescribed performance, *International Journal of Innovative Computing, Information and Control* 13 (2) (2017) 687–694.
- 415 [12] M. Benosman, K. . Lum, Passive actuators’ fault-tolerant control for affine nonlinear systems, *IEEE Transactions on Control Systems Technology* 18 (1) (2010) 152–163. doi:10.1109/TCST.2008.2009641.
- [13] Y. Yang, D. Yue, Observer-based decentralized adaptive nns fault-tolerant control of a class of large-scale uncertain nonlinear systems with actuator failures, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 49 (3) (2019) 528–542. doi:10.1109/TSMC.2017.2744676.
- 420 [14] Y. Song, X. Huang, C. Wen, Robust adaptive fault-tolerant pid control of mimo nonlinear systems with unknown control direction, *IEEE Transactions on Industrial Electronics* 64 (6) (2017) 4876–4884. doi:10.1109/TIE.2017.2669891.
- 425 [15] S. Tong, T. Wang, Y. Li, Fuzzy adaptive actuator failure compensation control of uncertain stochastic nonlinear systems with unmodeled dynamics, *IEEE Trans-*

- actions on Fuzzy Systems 22 (3) (2014) 563–574. doi:10.1109/TFUZZ.2013.2264939.
- 430 [16] E. Kamal, A. Aitouche, R. Ghorbani, M. Bayart, Robust fuzzy fault-tolerant control of wind energy conversion systems subject to sensor faults, IEEE Transactions on Sustainable Energy 3 (2) (2012) 231–241. doi:10.1109/TSTE.2011.2178105.
- [17] Y. Song, J. Guo, Neuro-adaptive fault-tolerant tracking control of lagrange systems pursuing targets with unknown trajectory, IEEE Transactions on Industrial
435 Electronics 64 (5) (2017) 3913–3920. doi:10.1109/TIE.2016.2644606.
- [18] H. Alwi, C. Edwards, Fault detection and fault-tolerant control of a civil aircraft using a sliding-mode-based scheme, IEEE Transactions on Control Systems
Technology 16 (3) (2008) 499–510. doi:10.1109/TCST.2007.906311.
- 440 [19] S. Islam, X. P. Liu, Robust sliding mode control for robot manipulators, IEEE Transactions on Industrial Electronics 58 (6) (2011) 2444–2453. doi:10.1109/TIE.2010.2062472.
- [20] Y. Chang, Adaptive sliding mode control of multi-input nonlinear systems with perturbations to achieve asymptotical stability, IEEE Transactions on Automatic
445 Control 54 (12) (2009) 2863–2869. doi:10.1109/TAC.2009.2033748.
- [21] M. Liu, L. Zhang, P. Shi, Y. Zhao, Fault estimation sliding-mode observer with digital communication constraints, IEEE Transactions on Automatic Control 63 (10) (2018) 3434–3441. doi:10.1109/TAC.2018.2794826.
- [22] Q. Xu, Continuous integral terminal third-order sliding mode motion control for piezoelectric nanopositioning system, IEEE/ASME Transactions on Mechatronics
450 22 (4) (2017) 1828–1838. doi:10.1109/TMECH.2017.2701417.
- [23] T. Madani, B. Daachi, K. Djouani, Modular-controller-design-based fast terminal sliding mode for articulated exoskeleton systems, IEEE Transactions on Control Systems Technology 25 (3) (2017) 1133–1140. doi:10.1109/TCST.2016.
455 2579603.

- [24] C. U. Solis, J. B. Clempner, A. S. Poznyak, Fast terminal sliding-mode control with an integral filter applied to a van der pol oscillator, *IEEE Transactions on Industrial Electronics* 64 (7) (2017) 5622–5628. doi:10.1109/TIE.2017.2677299.
- 460 [25] C. Chen, S. S. Xu, Y. Liang, Study of nonlinear integral sliding mode fault-tolerant control, *IEEE/ASME Transactions on Mechatronics* 21 (2) (2016) 1160–1168. doi:10.1109/TMECH.2015.2474700.
- [26] M. Van, S. S. Ge, H. Ren, Finite time fault tolerant control for robot manipulators using time delay estimation and continuous nonsingular fast terminal sliding mode control, *IEEE Transactions on Cybernetics* 47 (7) (2017) 1681–1693. doi:10.1109/TCYB.2016.2555307.
- 465 [27] M. Van, X. P. Do, M. Mavrovouniotis, Self-tuning fuzzy pid-nonsingular fast terminal sliding mode control for robust fault tolerant control of robot manipulators, *ISA Transactions* doi:https://doi.org/10.1016/j.isatra.2019.06.017. URL <http://www.sciencedirect.com/science/article/pii/S0019057819302782>
- [28] V. Utkin, Discussion aspects of high-order sliding mode control, *IEEE Transactions on Automatic Control* 61 (3) (2016) 829–833. doi:10.1109/TAC.2015.2450571.
- 475 [29] J. Zhang, X. Liu, Y. Xia, Z. Zuo, Y. Wang, Disturbance observer-based integral sliding-mode control for systems with mismatched disturbances, *IEEE Transactions on Industrial Electronics* 63 (11) (2016) 7040–7048. doi:10.1109/TIE.2016.2583999.
- [30] Y. Cao, X. B. Chen, Disturbance-observer-based sliding-mode control for a 3-dof nanopositioning stage, *IEEE/ASME Transactions on Mechatronics* 19 (3) (2014) 924–931. doi:10.1109/TMECH.2013.2262802.
- 480 [31] G. Rubio-Astorga, J. D. Snchez-Torres, J. Caedo, A. G. Loukianov, High-order sliding mode block control of single-phase induction motor, *IEEE Transactions*

- on Control Systems Technology 22 (5) (2014) 1828–1836. doi : 10.1109/TCST.2013.2289307.
- 485
- [32] A. Levant, Robust exact differentiation via sliding mode technique, *Automatica* 34 (3) (1998) 379 – 384.
- [33] Y. Feng, F. Han, X. Yu, Chattering free full-order sliding-mode control, *Automatica* 50 (4) (2014) 1310 – 1314.
- 490 [34] R. Errouissi, A. Al-Durra, S. M. Mueeen, Design and implementation of a nonlinear pi predictive controller for a grid-tied photovoltaic inverter, *IEEE Transactions on Industrial Electronics* 64 (2) (2017) 1241–1250. doi : 10.1109/TIE.2016.2618339.
- [35] C. F. de Paula, L. H. C. Ferreira, An improved analytical pid controller design for non-monotonic phase lti systems, *IEEE Transactions on Control Systems Technology* 20 (5) (2012) 1328–1333. doi : 10.1109/TCST.2011.2163636.
- 495
- [36] Y. I. Son, I. H. Kim, Complementary pid controller to passivity-based nonlinear control of boost converters with inductor resistance, *IEEE Transactions on Control Systems Technology* 20 (3) (2012) 826–834. doi : 10.1109/TCST.2011.2134099.
- 500
- [37] L. Hao, J. H. Park, D. Ye, Integral sliding mode fault-tolerant control for uncertain linear systems over networks with signals quantization, *IEEE Transactions on Neural Networks and Learning Systems* 28 (9) (2017) 2088–2100. doi : 10.1109/TNNLS.2016.2574905.
- [38] C.-S. Chiu, Derivative and integral terminal sliding mode control for a class of mimo nonlinear systems, *Automatica* 48 (2) (2012) 316 – 326.
- 505
- [39] M. Van, S. S. Ge, H. Ren, Robust fault-tolerant control for a class of second-order nonlinear systems using an adaptive third-order sliding mode control, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 47 (2) (2017) 221–228. doi : 10.1109/TSMC.2016.2557220.
- 510

- [40] Z. Wang, C. Hu, Y. Zhu, S. He, K. Yang, M. Zhang, Neural network learning adaptive robust control of an industrial linear motor-driven stage with disturbance rejection ability, *IEEE Transactions on Industrial Informatics* 13 (5) (2017) 2172–2183. doi:10.1109/TII.2017.2684820.
- 515 [41] Z. Liu, X. Dong, J. Xue, H. Li, Y. Chen, Adaptive neural control for a class of pure-feedback nonlinear systems via dynamic surface technique, *IEEE Transactions on Neural Networks and Learning Systems* 27 (9) (2016) 1969–1975. doi:10.1109/TNNLS.2015.2462127.
- [42] C. Hwang, C. Jan, Recurrent-neural-network-based multivariable adaptive control for a class of nonlinear dynamic systems with time-varying delay, *IEEE Transactions on Neural Networks and Learning Systems* 27 (2) (2016) 388–401. doi:10.1109/TNNLS.2015.2442437.
- 520
- [43] S. Wen, M. Z. Q. Chen, Z. Zeng, T. Huang, C. Li, Adaptive neural-fuzzy sliding-mode fault-tolerant control for uncertain nonlinear systems, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 47 (8) (2017) 2268–2278. doi:10.1109/TSMC.2017.2648826.
- 525
- [44] W. He, Y. Dong, Adaptive fuzzy neural network control for a constrained robot using impedance learning, *IEEE Transactions on Neural Networks and Learning Systems* 29 (4) (2018) 1174–1186. doi:10.1109/TNNLS.2017.2665581.
- [45] K. S. Tang, Kim Fung Man, Guanrong Chen, S. Kwong, An optimal fuzzy pid controller, *IEEE Transactions on Industrial Electronics* 48 (4) (2001) 757–765. doi:10.1109/41.937407.
- 530
- [46] M. J. Neath, A. K. Swain, U. K. Madawala, D. J. Thrimawithana, An optimal pid controller for a bidirectional inductive power transfer system using multiobjective genetic algorithm, *IEEE Transactions on Power Electronics* 29 (3) (2014) 1523–1531. doi:10.1109/TPEL.2013.2262953.
- 535
- [47] C. Liu, Y. Hsu, Design of a self-tuning pi controller for a statcom using particle

- swarm optimization, *IEEE Transactions on Industrial Electronics* 57 (2) (2010) 702–715. doi:10.1109/TIE.2009.2028350.
- 540 [48] W. Sa-ngiamvibool, Optimal fuzzy logic proportional integral derivative controller design by bee algorithm for hydro-thermal system, *IEEE Transactions on Industrial Informatics* (2017) 1–1doi:10.1109/TII.2017.2647812.
- [49] X. Yang, Bat algorithm: a novel approach for global engineering optimization, *Engineering Computations* 29 (5) (2012) 464483.
- 545 [50] H. Huang, Fusion of modified bat algorithm soft computing and dynamic model hard computing to online self-adaptive fuzzy control of autonomous mobile robots, *IEEE Transactions on Industrial Informatics* 12 (3) (2016) 972–979. doi:10.1109/TII.2016.2542206.
- [51] Z.-W. Ye, M.-W. Wang, W. Liu, S.-B. Chen, Fuzzy entropy based optimal thresholding using bat algorithm, *Applied Soft Computing* 31 (2015) 381 – 395.
- 550 [52] N. S. Jaddi, S. Abdullah, A. R. Hamdan, Multi-population cooperative bat algorithm-based optimization of artificial neural network model, *Information Sciences* 294 (2015) 628 – 644, innovative Applications of Artificial Neural Networks in Engineering.
- 555 [53] B. Jiang, M. Staroswiecki, V. Cocquempot, Fault accommodation for nonlinear dynamic systems, *IEEE Transactions on Automatic Control* 51 (9) (2006) 1578–1583. doi:10.1109/TAC.2006.878732.
- [54] Q. Yang, S. Jagannathan, Y. Sun, Robust integral of neural network and error sign control of mimo nonlinear systems, *IEEE Transactions on Neural Networks and Learning Systems* 26 (12) (2015) 3278–3286. doi:10.1109/TNNLS.2015.2470175.
- 560 [55] H. Wang, X. Liu, K. Liu, H. R. Karimi, Approximation-based adaptive fuzzy tracking control for a class of nonstrict-feedback stochastic nonlinear time-delay systems, *IEEE Transactions on Fuzzy Systems* 23 (5) (2015) 1746–1760. doi:10.1109/TFUZZ.2014.2375917.
- 565

- [56] X. Zhao, H. Yang, H. R. Karimi, Y. Zhu, Adaptive neural control of mimo nonstrict-feedback nonlinear systems with time delay, *IEEE Transactions on Cybernetics* 46 (6) (2016) 1337–1349. doi:10.1109/TCYB.2015.2441292.
- [57] Y. Liang, S. Xu, C. Tsai, Study of vsc reliable designs with application to spacecraft attitude stabilization, *IEEE Transactions on Control Systems Technology* 15 (2) (2007) 332–338. doi:10.1109/TCST.2006.883186.
- [58] S. K. Kommuri, M. Defoort, H. R. Karimi, K. C. Veluvolu, A robust observer-based sensor fault-tolerant control for pmsm in electric vehicles, *IEEE Transactions on Industrial Electronics* 63 (12) (2016) 7671–7681. doi:10.1109/TIE.2016.2590993.