



ROBUST FAULT TOLERANT CONTROL BASED ON LADRC FOR THE QUADROTOR*

JIAJIN LI, RUI LI[†], YINGJING SHI AND LIANG HE

Abstract: This paper focuses on the position and attitude control for a quadrotor having disturbance and actuator faults during the flight. The quadrotor is an underactuated, strong coupled, nonlinear complex system with uncertain parameters and particularly sensitive to external disturbance which is difficult to model. In addition, partial failure of four actuators of the quadrotor is the most frequently occurred in the actual flight test. In order to solve the above problems, a robust fault tolerant controller is designed in this paper that combines time-delay control (TDC) and the linear adaptive disturbance rejection control (LADRC) method. The performance of the controller is analyzed by simulation experiments, which show that the designed controller is with good performance for suppressing parameter uncertainties, external disturbances and fault tolerant of actuator faults.

Key words: *quadrotor, modelling, linear adaptive disturbance rejection control (LADRC), fault tolerant control, time delay control*

Mathematics Subject Classification: *93C85, 70Q05, 93C73*

1 Introduction

In recent years, the quadrotor has attracted wild attention of scientists and engineers due to the advantages of vertically takeoff and landing, hovering, cruising, rapidly changing course and good agility compared with the fixed wing UAVs [14]. In addition, the quadrotor has also been widely used in military and civil applications such as low-altitude surveillance, security patrols, rescue and aerial photography [4, 11, 19, 25].

When the quadrotor flies in outdoor environments for a long time, its actuators are prone to ageing and can be damaged by collisions. Therefore, it is frequently happened that four actuators get stuck and become partial failure in actual flight tests. The actuator fault is a security threat to the quadrotor flight control system. In addition, the quadrotor is a strongly coupled, nonlinear, underactuated complex system and its flight control system only has four control inputs, so it is particularly sensitive to external disturbances. How to achieve stable flight of the quadrotor in the presence of actuator faults and external disturbance is an urgent problem to be solved.

*This work is supported in part by the National Natural Science Foundation of China under grant (No. 61973055), the Fundamental Research Funds for the Central Universities (No. ZYGX2019J062), and a grant from the applied basic research programs of Sichuan province (No. 2019YJ0206).

[†]Corresponding author.

A robust and highly reliable control system of the quadrotor must have the following characteristics:

1. The flight control system must have good robustness to deal with the uncertainty of the quadrotor model's parameters and external disturbance.
2. The flight control system can ensure flight safety by corrections of control inputs when the quadrotor has actuator faults.

The researchers use sliding mode disturbance observer (SMC-SMDO) to design a robust flight controller when the quadrotor faces with the external disturbances and model uncertainties in [3]. Backstepping based control method is proposed in [12] to design a nonlinear adaptive controller which can compensate for the mass uncertainty of the quadrotor. Researchers have done a lot of research on how to achieve a robust and highly reliable quadrotor flight control system. However, there is few work which considers both the disturbance and actuator faults at the same time. Therefore, a LADRC fault tolerant control system is developed for this situation.

Active disturbance rejection control(ADRC) proposed by Jingqing Han [21] is an effective nonlinear control algorithm which combines the advantages of the traditional PID algorithm and the analysis method of modern control theory. ADRC has attracted extensive attention from experts and scholars in the field of control since it was proposed. ADRC has been proven as a robust control method for the quadrotor which can estimate the disturbances and eliminate the effects in time [16, 18, 22, 30]. But the main drawback of the ADRC that is many parameters need to be tuned. Then, Dr. Gao Zhiqiang proposes the linear active disturbance rejection control (LADRC) [9, 10] which inherits the strong robustness of ADRC and has fewer parameters to be chosen. These advantages make the algorithm easy to be implemented in engineering applications.

Fault tolerant control technology is a kind of advanced control method that can automatically compensate for the impact of faults when the system has a certain degree of faults [24]. Many fault tolerant control methods of the quadrotor require estimating the failure information on-line and in real time [6, 23, 29]. However, due to the existence of various noise, the fault information estimated by the observer may not be correct during the actual process, which may reduce the system performance and even result unstable. This paper designs the LADRC fault tolerant controller that combines time-delay control (TDC) and LADRC method. The method uses LADRC to estimate the inertia uncertainty and external disturbance with the help of the extended state observer. At the same time, the time-delay control [5, 32] is used to compensate for the actuator faults. The proposed method does not require on-line fault detection and separation. The unknown actuator faults are estimated by using one-step iteration of the previous state information and canceled out by the estimated values. This designed control algorithm can not only eliminate the effects of disturbances but also ensure the safety of the quadrotor in the event of actuators faults.

The structure of this paper is organized as follows. In section 2, the mathematical model of the quadrotor is established and is linearized respectively for the altitude model, the attitude model and the planar motion model. In section 3, the robust fault tolerant control strategy of the quadrotor is developed which combines robust flight control algorithm LADRC and passive fault tolerant algorithm TDC. In Section 4, simulation studies are carried out to verify the effectiveness of the proposed algorithm.

2 Quadrotor Dynamic Model

The structure of oblique cross quadrotor is presented as Figure 1. As the design of the quadrotor has ensured that its structure is basic symmetry, the quadrotor can be regarded

as a rigid body when the flight speed and wind speed are low. The dynamic model of the quadrotor can be obtained by using Lagrangian equation and Newton's second law, which is described by the nonlinear system of twelve ordinary differential equations [17, 20, 27]:

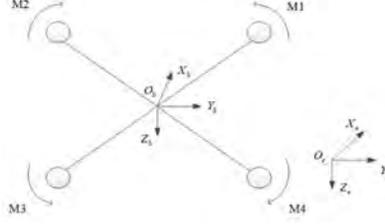


Figure 1: The structure of oblique cross quadrotor

$$\dot{X} = f(X, U) + W \quad (2.1)$$

where $X = (x_1 \cdots x_{12})^T \in \mathbb{R}^{12}$ is the state vector including the rotational components $(\phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi})^T$ (ϕ -roll angle, θ -pitch angle, ψ -yaw angle) and the translational components $(x, \dot{x}, y, \dot{y}, z, \dot{z})^T$ of the quadrotor; $U = (U_1, U_2, U_3, U_4)$ is the control input vector; $W = [0, W_1 \cdots 0, W_6]^T$ is the disturbance vector caused by the aerodynamic forces. The equation (2.1) can be expressed as:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \\ \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{pmatrix} = \begin{pmatrix} \dot{\phi} \\ \ddot{\phi} \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{\psi} \\ \ddot{\psi} \\ \dot{x} \\ \ddot{x} \\ \dot{y} \\ \ddot{y} \\ \dot{z} \\ \ddot{z} \end{pmatrix} = \begin{pmatrix} x_2 \\ a_1 x_4 x_6 + a_2 \Omega_r x_4 + b_1 U_2 \\ x_4 \\ a_3 x_2 x_6 - a_4 \Omega_r x_2 + b_2 U_3 \\ x_6 \\ a_5 x_2 x_4 + b_3 U_4 \\ x_8 \\ u_x U_1 / m \\ x_{10} \\ u_y U_1 / m \\ x_{12} \\ g - (\cos x_3 \cos x_1) U_1 / m \end{pmatrix} + \begin{pmatrix} 0 \\ W_1 \\ 0 \\ W_2 \\ 0 \\ W_3 \\ 0 \\ W_4 \\ 0 \\ W_5 \\ 0 \\ W_6 \end{pmatrix} \quad (2.2)$$

$$\begin{aligned} a_1 &= (J_y - J_z) / J_x, a_3 = (J_z - J_x) / J_y, a_5 = (J_x - J_y) / J_z, \\ a_2 &= J_{rz} / J_x, a_4 = J_{rz} / J_y, a_6 = J_{rz} / J_z, \\ b_1 &= 1 / J_x, b_2 = 1 / J_y, b_3 = 1 / J_z, \\ \Omega_r &= \omega_1 - \omega_2 + \omega_3 - \omega_4, \\ u_x &= \cos x_1 \sin x_3 \cos x_5 + \sin x_1 \sin x_5, \\ u_y &= \cos x_1 \sin x_3 \sin x_5 - \sin x_1 \cos x_5. \end{aligned} \quad (2.3)$$

Here $\omega_i, i = 1, 2, 3, 4$ are the revolutions of four rotors respectively; J_x, J_y, J_z denote the moment of inertia of three-axis in the body coordinate system; J_{rz} is the inertia of rotors; Ω_r is the overall residual rotors angular speed; the control input U_1 is the total lift force produced by the rotors and the control inputs U_2, U_3, U_4 are the torque of the quadrotor

about the three axis which are given below:

$$\begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{pmatrix} = \begin{pmatrix} F \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{pmatrix} = \begin{pmatrix} b & b & b & b \\ bl & -bl & -bl & bl \\ -bl & -bl & bl & bl \\ -d & d & -d & d \end{pmatrix} \begin{pmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{pmatrix} = P_A \omega \quad (2.4)$$

where parameter l expresses the distance between the rotor and the centre of the quadrotor; the parameters b and d are the lift coefficient and the torque coefficient of four rotors respectively.

To facilitate the description of the fault-tolerant algorithm mentioned in the next section, we need to decouple and linearize the model of the quadrotor. Since the angles ϕ, θ, ψ are independent of the translational components, the system's attitude dynamics in (2.2) can be decoupled from the translational ones as in [1]. As shown in (2.2) altitude and planar motions can also be decoupled. Thus three different system models will be derived which are the attitude model, the altitude model and the planar motion model.

The attitude model can be obtained as:

$$\begin{pmatrix} \dot{\tilde{\phi}} \\ \ddot{\tilde{\phi}} \\ \dot{\tilde{\theta}} \\ \ddot{\tilde{\theta}} \\ \dot{\tilde{\psi}} \\ \ddot{\tilde{\psi}} \end{pmatrix} = \begin{pmatrix} \tilde{\phi} \\ a_1 \tilde{\theta} \dot{\tilde{\psi}} + a_2 \Omega_r \dot{\tilde{\theta}} \\ \tilde{\theta} \\ a_3 \tilde{\phi} \dot{\tilde{\psi}} - a_4 \Omega_r \dot{\tilde{\phi}} \\ \tilde{\psi} \\ a_5 \tilde{\theta} \dot{\tilde{\phi}} \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & b_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & b_2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b_3 \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{pmatrix} \quad (2.5)$$

where $\tilde{\phi}, \tilde{\theta}, \tilde{\psi}$ express the quadrotor's attitude angles θ with small attitude disturbance around the operating points.

The altitude model ($X_{Ez} = (\tilde{z}(t), z(\tilde{t}), \int \tilde{z}(t) dt)^T$) can be obtained by transforming the altitude dynamics into error dynamics and discretizing it with a T_s sampling time:

$$X_{Ez}(k+1) = \begin{pmatrix} 1 & T_s & 0 \\ 1 & 1 & 0 \\ T_s & 1 & 1 \end{pmatrix} X_{Ez}(k) + \begin{pmatrix} 0 \\ \frac{T_s}{m} \cos \tilde{\theta} \cos \tilde{\phi} \\ 0 \end{pmatrix} (\delta U_1) \quad (2.6)$$

Similar to the altitude model, the planar motion model $X_{ExEy} = (\tilde{x}(t), x(\tilde{t}), \int \tilde{x}(t) dt, \tilde{y}(t), y(\tilde{t}), \int \tilde{y}(t) dt)^T$ can be obtained as:

$$X_{ExEy}(k+1) = \begin{pmatrix} 1 & T_s & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ T_s & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T_s & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & T_s & 0 & 1 \end{pmatrix} X_{ExEy}(k) + \begin{pmatrix} 0 \\ \frac{T_s U_1(k)}{m} \\ 0 \\ 0 \\ \frac{T_s U_1(k)}{m} \\ 0 \end{pmatrix} \begin{pmatrix} u_x \\ u_y \end{pmatrix} \quad (2.7)$$

The parameters of quadrotor model are shown in Table I [7, 26]. We can use the parameters C_T and C_M to figure out the lift coefficient b and the torque coefficient d .

Table 1 THE QUADROTOR MODEL PARAMETERS

Parameter	Value	Unit
m	1.731	K_g
g	9.7913	m/s^2
l	0.225	m
C_T	1.0792×10^{-5}	$N \cdot s^2/rad^2$
C_M	1.8992×10^{-7}	$N \cdot m \cdot s^2/rad^2$
J_x	9.5065×10^{-3}	$K_g \cdot m^2$
J_y	1.0000×10^{-2}	$K_g \cdot m^2$
J_z	1.6580×10^{-2}	$K_g \cdot m^2$
J_{rz}	6.0000×10^{-5}	$K_g \cdot m^2$
T_s	0.01	s

From (2.2), we notice that there are some unknown parameters in the model of the quadrotor, where measurement errors will be brought when measuring these parameters. In addition, the disturbance vector caused by the aerodynamic forces will also affect the accuracy of the model. Furthermore, the four control inputs of the quadrotor will change when the quadrotor has actuator faults. Therefore, it is necessary to develop a reliable control system ensuring the safe flight of the quadrotor.

3 Robust Fault Tolerant Control System

In this section, the robust fault tolerant control strategy is used to design the flight controller of the quadrotor. The goal of the control strategy is to develop a robust fault tolerant system that can not only reject the uncertainty and external disturbance but also ensure flight safety. The robust fault tolerant control system mainly includes two parts: the robust control module and the fault processor module. The structural diagram of the robust fault tolerant control system of the quadrotor is shown in the figure below, in which the control inputs and outputs are position and attitude references.

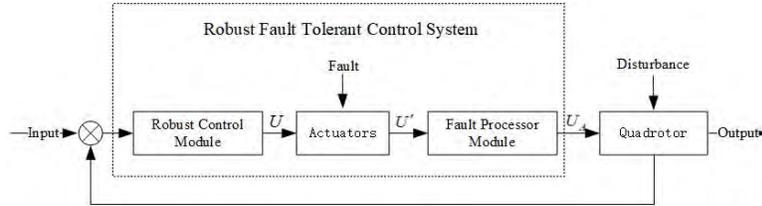


Figure 2: The structure of the robust fault tolerance control system

Here U is the control variable which is the output of the robust flight controller and the input of the actuator; U' is the output of U through the faulty actuators; U_A is the control variable to the quadrotor which is obtained by the robust fault tolerant control system.

3.1 Robust control module

In this module, a robust controller based on the LADRC is introduced to ensure stable flight of the quadrotor with the uncertainty of the model's parameters and external disturbance. We apply the LADRC in both the position control and the attitude control at the same time. The structure of the robust fault tolerant system is shown below:

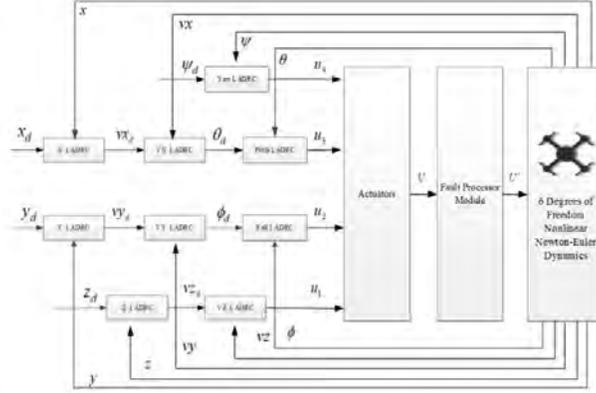


Figure 3: The structure of robust fault tolerant system based on LADRC

As shown in figure 3, the LADRC is applied to each loop separately. The linear extended state observer can estimate and compensate for the overall disturbance of the system in real time [8]. Considering that yaw angle ψ is generally maintained in a small range close to zero, we regard pitch angle θ as the attitude control input of the position x and roll angle ϕ as the attitude control input of the position y . In order to achieve better control performance, the designed LADRC position controller is divided into two loops: the outer loop controls the position and the inner loop controls the speed. Here the speed control can prevent excessive overshoot.

Each LADRC controller is composed of a linear extended state observer (LESO) and a linear controller. For brevity, we only show the design process of the LADRC controller for pitch channel. From the model equation (2.2), we can see the pitch channel is a second order system. We rearrange this second order system as follows:

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = z_3 + b_0 u \\ \dot{z}_3 = h \\ y = z_1 \end{cases} \quad (3.1)$$

where z_1 denotes pitch angle θ and z_2 denotes $\dot{\theta}$. h denotes the total disturbance including parameter uncertainty and the impact of the aerodynamic forces. The LADRC controller is divided into the following two parts.

(1) The linear extended state observer (LESO) is expressed as:

$$\begin{cases} e = \hat{z}_1 - y \\ \dot{\hat{z}}_1 = \hat{z}_2 - \beta_1 e \\ \dot{\hat{z}}_2 = \hat{z}_3 - \beta_2 e + b_0 u \\ \dot{\hat{z}}_3 = -\beta_3 e \end{cases} \quad (3.2)$$

where $\hat{z} = [\hat{z}_1, \hat{z}_2, \hat{z}_3]$ is the estimated state vector of the state vector $z = [z_1, z_2, z_3]$. b_0 is the estimation value of b . Since b is unknown, b_0 needs to be tuned. $L = [\beta_1, \beta_2, \beta_3]^T$ is the observer gain vector which can be selected appropriately by using the pole-placement method. In addition, to simplify the process of tuning parameter, we assume that the characteristic polynomial of (3.2) is

$$\lambda(s) = s^3 + \beta_1 s^2 + \beta_2 s + \beta_3 = (s + \omega_0)^3 \quad (3.3)$$

Hence, the observer gains are parameterized as

$$\begin{cases} \beta_1 = 3\omega_0 \\ \beta_2 = 3\omega_0^2 \\ \beta_3 = \omega_0^3 \end{cases} \quad (3.4)$$

where ω_0 is related to the observer bandwidth.

(2) Then, the control system is reduced to a cascaded integrator by using the following simple PD control law:

$$u_0 = k_p(r - z_1) - k_d z_2 - z_3 \quad (3.5)$$

The value of the gains k_p and k_d can be tuned by placing all the closed-loop poles at $-\omega_c$, where ω_c is derived by the bandwidth of the feedback control system. Furthermore, we can obtain $k_p = \omega_c^2$, $k_d = 2\omega_c$ [10].

3.2 Parameter optimization of the control system

According to the previous section, we need to adjust three parameters of the LADRC controller: the controller bandwidth ω_c , the observer bandwidth ω_0 , and the compensation factor b_0 . If the structure of the control system keeps unchanged, its control performance generally depends on the choice of parameters. Therefore, we need to optimize the parameters to achieve a better performance. The particle swarm optimization (PSO) [2, 15, 31] technique is used for parameters tuning. Considering that the standard PSO has some shortcomings such as easily falling into the local extremum point, slow convergence speed at the end of the algorithm iteration and low accuracy [28, 33], this paper introduces a new intelligent hybrid algorithm combining the PSO algorithm with the effective steepest descent method. Taking the channel $x - \theta$ as an example, the parameters of the LADRC controller of the position x , the LADRC controller of the velocity v_x and the LADRC controller of the pitch are jointly adjusted by the hybrid PSO. The process is shown in figure 4.

The LADRC parameters optimization process is shown in figure 5.

The steps of the hybrid optimization algorithm proposed in this paper is as follows:

(1) Initialize the population and set the maximum number of iterations of the particle swarm, the maximum iteration number of steepest descent or the iterative accuracy of steepest descent.

(2) Use particle swarm algorithm to generate population and calculate the best historical point of each particle P_{best} and the best point of the entire population P_g .

(3) The steepest descent method is used to search the extreme point P'_g with the best point P_g of the entire population. And then regard this extreme point as P_g (As P_g may not be updated in many generations, we can set the flag that P_g has been updated in the program design, and the fastest search only done for the updated P_g).

(4) Update the speed V_{id} according to the velocity update formula of particle swarm optimization.

(5) Update the position X_{id} according to $X_{id}^{k+1} = X_{id}^k + V_{id}^{k+1}$, return to step (2) and continue the PSO iteration until the given number of iterations is reached.

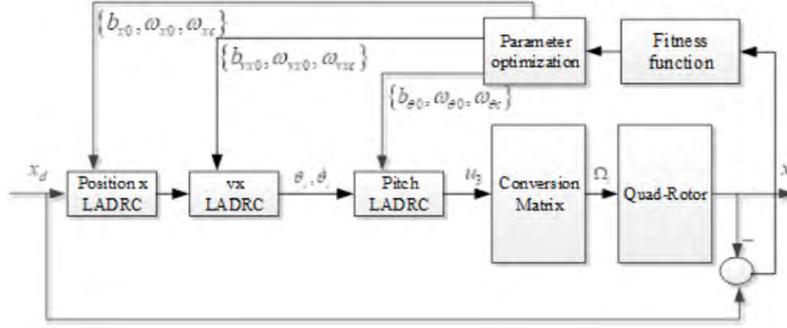


Figure 4: The structure of the parameters optimization

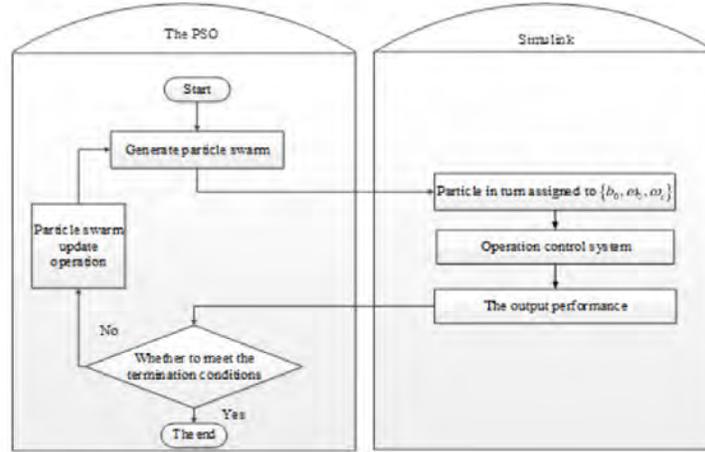


Figure 5: The process of the parameters optimization

3.3 Fault processor module

We apply the time-delay control algorithm to design the fault tolerant controller which enable the quadrotor to maintain stable flight when the quadrotor has actuator faults. Time-delay control is an effective method to deal with system faults [33]. It can approach fault information through one-step state iteration. This section discusses the impact on the quadrotor when a single actuator loses its thrust partially, where the fault of the actuator is within an acceptable range. When the actuator of the quadrotor loses its thrust partially, the actuators motor voltage output becomes lower, and the input of the system will change. However, the other structure of the system maintains the same. The actuator faults do not affect the mathematical model established in section II.

Here we use the attitude model as an example to introduce the fault tolerant control algorithm. When the quadrotor has actuator faults, the inputs of the quadrotor will change. According to the equation (2.4), the inputs with the actuator faults can be expressed as:

$$U' = PU = PP_A\omega \quad (3.6)$$

where $P = \text{diag}(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ is the fault coefficient matrix; α_i is the effective factor of the actuator.

According to the equation (2.5), the attitude model can be expressed as:

$$\begin{aligned} \dot{X} &= \begin{pmatrix} \dot{\phi} \\ a_1 \dot{\theta} \dot{\psi} + a_2 \Omega_r \dot{\theta} \\ \dot{\theta} \\ a_3 \dot{\phi} \dot{\psi} - a_4 \Omega_r \dot{\phi} \\ \dot{\psi} \\ a_5 \dot{\theta} \dot{\phi} \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & b_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & b_2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b_3 \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{pmatrix} \\ &= f + BP_A P \omega \\ &= f(x) + BU' \end{aligned} \quad (3.7)$$

Transform the above formula as follows:

$$\dot{X} = f + BU_A + BU' - BU_A \quad (3.8)$$

Let

$$g = BU' - BU_A \quad (3.9)$$

Hence, the equation (3.8) can be expressed as:

$$\dot{X} = f + BU_A + g \quad (3.10)$$

Considering the actuator fault, U_A is designed as

$$U_A = B^+(U - g) \quad (3.11)$$

where B^+ is the generalized inverse of the matrix B . When the actuator has no fault, the fault coefficient matrix can be expressed as $P = \text{diag}(1, 1, 1, 1)$. Thus, the fault tolerant controller is automatically converted into the LADRC controller.

However, the value of the fault coefficient matrix P cannot be obtained when the actuator is with fault, thus g cannot be determined. When the quadrotor has the actuator faults during a stable flight, neither the revolutions of the rotor can not change suddenly nor the attitude of the quadrotor. Therefore, the control law (3.11) can be implemented by using the estimated value \hat{g} instead of g . \hat{g} is defined as:

$$\hat{g} = g(t - T_s) = \dot{x}(t - T_s) - f(t - T_s) - BU_d^f(t - T_s) \quad (3.12)$$

So, the input of the quadrotor can be expressed as:

$$U_A = B^+(U - \hat{g}(t - T_s) + f(t - T_s) + BU_A(t - T_s)) \quad (3.13)$$

where

$$\hat{g}(t - T) = \frac{x(t - T) - x(t - 2T)}{T} \quad (3.14)$$

Remark 3.1. When the fault tolerant control algorithm is applied to the altitude model or the planar motion model, according to the formulas (2.6) and (2.7), we can see that

$$B = \begin{pmatrix} 0 \\ \frac{T_s}{m} \cos \tilde{\theta} \cos \tilde{\phi} \\ 0 \end{pmatrix} \text{ or } B = \begin{pmatrix} 0 & 0 \\ \frac{T_s U_1(k)}{m} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \frac{T_s U_1(k)}{m} \\ 0 & 0 \end{pmatrix}.$$

4 Simulation Analysis

In this section, simulation studies are carried out to verify the effectiveness of the designed LADRC fault tolerant control system of the quadrotor.

4.1 The attitude and position control simulation experiments without actuator faults

In attitude and position control tests, the control parameters of the LADRC controller can be obtained using the parameter optimization method discussed in section 3. The initial parameters of HPSO are set as shown in Table 2 and the final determined LADRC parameters are shown in Table 3.

TABLE 2 INITIALIZATION PARAMETERS SETTING FOR HPSO

Particle swarm size	Maximum iterations	Acceleration factor c_1	Acceleration factor c_2	Initial step size l
100	100	2	2	1.0

TABLE 3 PARAMETERS OPTIMIZATION RESULT FOR LADRC

Channel	LADRC		
	b	ω_0	ω_c
Position Z	2.24	1.56	5.06
vz	0.05	12.96	3.08
$x - vx - \theta$	1.56	0.65	109.86
$y - vy - \phi$	49.65	21.23	3.58
Yaw	0.89	0.76	23.42
	0.08	0.85	2.87

Through the optimization of the hybrid particle swarm optimization algorithm, the three parameters of each LADRC controller are optimized, which achieve the purpose of optimization the system performance. The figure 6 shows the yaw angle response curve before and after the optimization. The blue curve represents the yaw angle response curve before parameters optimization, and the red represents the curve of the yaw angle response curve after parameters optimization. It can be seen from the figure that the rate of convergence becomes faster and the fluctuating range becomes smaller after the parameters optimization.

The initial values of pitch angle, roll angle and yaw angle are set to 0 rad. We send the reference inputs, which are rectangular wave signals, to yaw angles, roll angles and pitch angles respectively at 3 s, 6 s, and 9 s. The output curves of the system are shown in the figure below. The results are compared with the PID controller.

Fig. 7(a) presents the simulation results of the LADRC fault tolerant controller. Fig. 4(b) presents the simulation results of PID controller. From the results, the attitude response curve of the LADRC controller can track the expectation smoothly with fast response and small overshoot which are respectively 10.02%, 7.56%, and 7.48%. Compared with PID, it can also be seen that when the three attitude angles change, there is almost no effect on other angles. This is because that the LESO of the LADRC controller can treat the coupling between the three angle channels as the internal disturbances of the system and compensates for the disturbances in real time.

Furthermore, we add 5% Gaussian white noise to the state feedback variables of the attitude angles. Then, we simulate the noise disturbance in the position control tests to

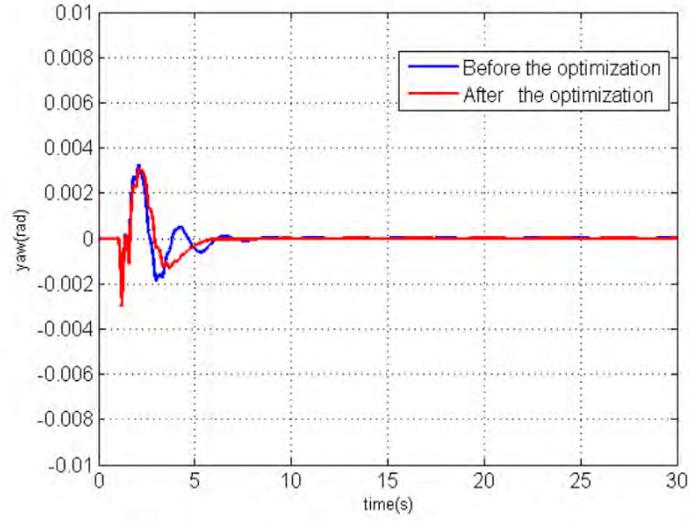


Figure 6: The yaw angle response curve before and after the optimization

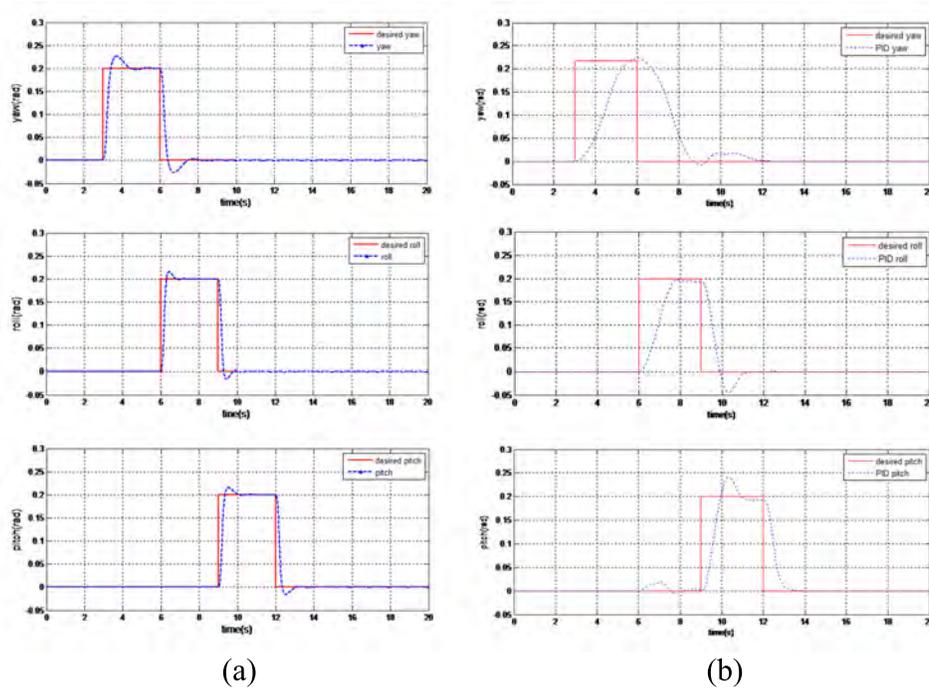


Figure 7: Attitude angle response curves

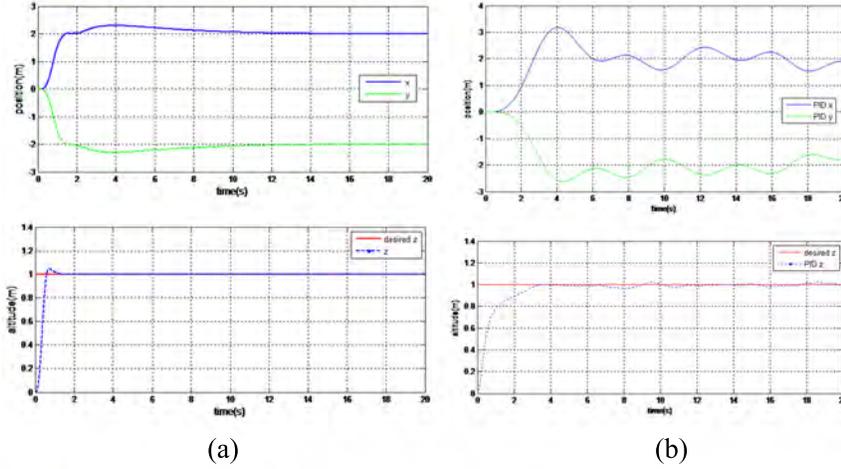


Figure 8: Position response curves with noise

verify the robustness of the controller. The tests make the quadrotor hover at position (2, -2) and reach a height of 1m. The system output curve is shown in Figure 8.

Fig. 8(a) presents the simulation results of the LADRC fault tolerant controller. Fig. 8(b) presents the simulation results of PID controller. From the results, the double closed-loop position control of LADRC can achieve hover control with fast response and small overshoot and has good robustness to overcome the effect of noise disturbance.

The following tests will analyze and verify the disturbance rejection and robustness of LADRC.

1) Analysis of disturbance rejection

5% Gaussian white noise and transient signals are added to the system's feedback state variables to test the disturbance rejection ability of the control system. Take the yaw angle as an example to analyze the system output.

Fig. 9(a) presents the simulation results of the LADRC fault tolerant controller. Fig. 9(b) presents the simulation results of PID controller. Fig. 9(c) presents the total disturbance estimated by the LESO. From the above figures, we can see that the quadrotor can return to stable state within 2s and can control the error below 0.07 rad when there exists disturbance. Compared with PID, LADRC fault tolerant control has better anti-disturbance ability.

2) Analysis of robustness

The mass of the quadrotor is firstly increased by 30%, and then the moment of inertia is reduced by 30%. The LADRC parameters are kept constant to test the robustness of the control system. Take the yaw angle as an example to show the tracking response under different parameters.

From the fig. 10, we can see that the three tracking curves basically coincide when the model parameters of the quadrotor change by 30%. Therefore, when the model parameters cannot be determined, the parameters of LADRC fault tolerant controller do not need to be adjusted significantly and the controller has strong robustness.

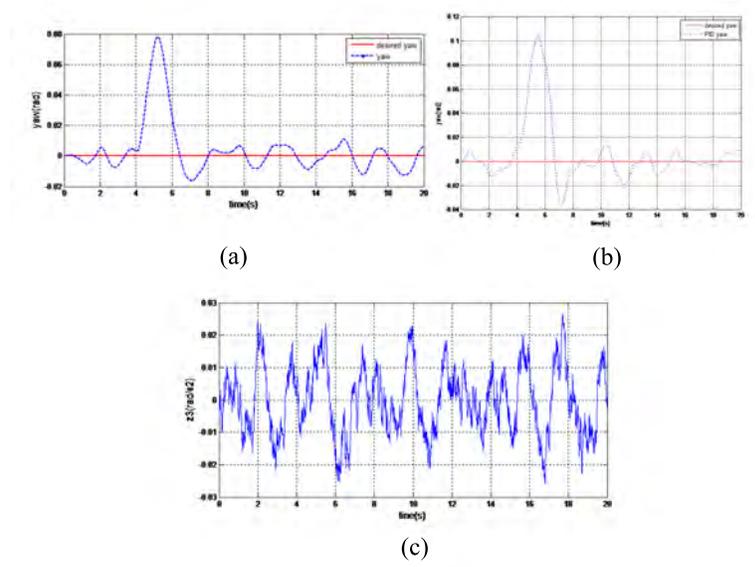


Figure 9: Outputs of attitude angle yaw and LESO

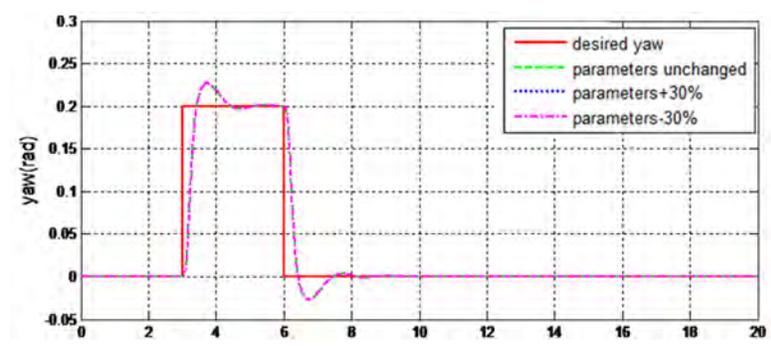


Figure 10: Outputs of attitude angle yaw with different parameters

4.2 The attitude and position control simulation experiments with actuator faults

The simulation experiment is carried out to verify the effectiveness of the fault tolerant control system with 20% loss failure of No.1 actuator. According to the previous analysis, $\alpha_1' = 0.8\alpha_1$ when the No.1 actuator has 20% loss failure. When the quadrotor is affected by 5% Gaussian white noise and has actuator failure at 20s, the system attitude and position outputs are shown in Figure 11.

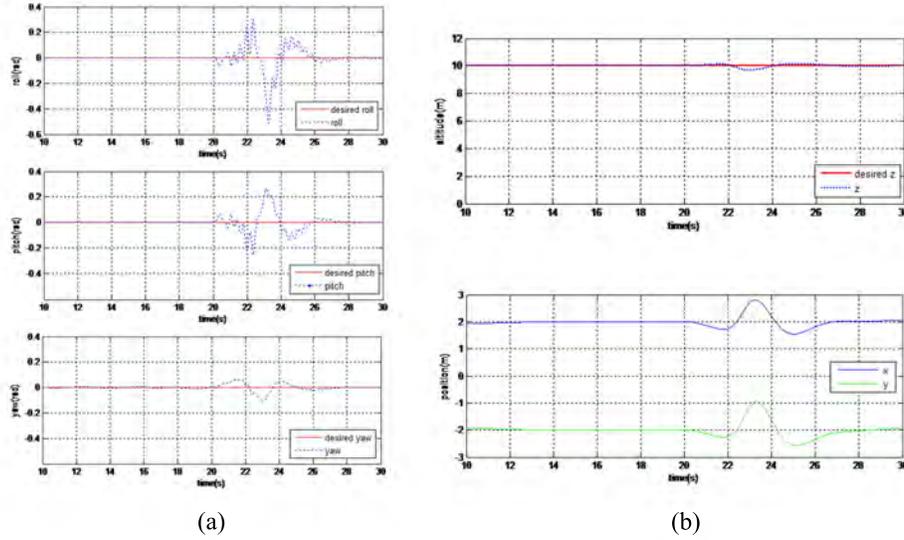


Figure 11: Attitude angle(a) and position(b) response curves with actuator faults and noise

The simulation results show that the LADRC fault tolerant controllers can effectively deal with actuator faults in attitude control and position control. From the figures, when the actuator fault occurs, fault tolerant control can make the system reach stable flight state within 8s. Although the adjustment time is slightly longer, the adjustment process is smooth, which can ensure that the quadrotor hover smoothly to the stable point.

After the quadrotor can achieve stabilized flight with an actuator fault when it is hovering, the tracking ability of fault tolerant control is further analyzed. Set the target tracking trajectory as an ellipse, which can be expressed by the following formula:

$$\begin{cases} x_d = 12 \sin(0.5t) \\ y_d = 6 \cos(0.5t) \\ z_d = 10 \end{cases} \quad (4.1)$$

Noise and actuator faults are the same as previous tests. The tracking performance and the control inputs of the quadrotor are shown in Figure 12 and Figure 13.

In figure 12, the blue curve presents the flight path with LADRC fault tolerant control and the green curve presents the flight path without fault tolerant control. Fig 13(a) presents the control inputs of the quadrotor with LADRC fault tolerant controller. Fig 13(b) presents the control inputs of the quadrotor with PID controller. Comparison of these two flight trajectories shows that the fault tolerance control based on LADRC and TDC can make the

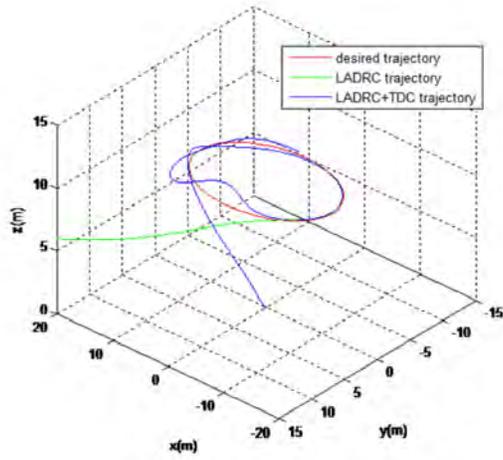
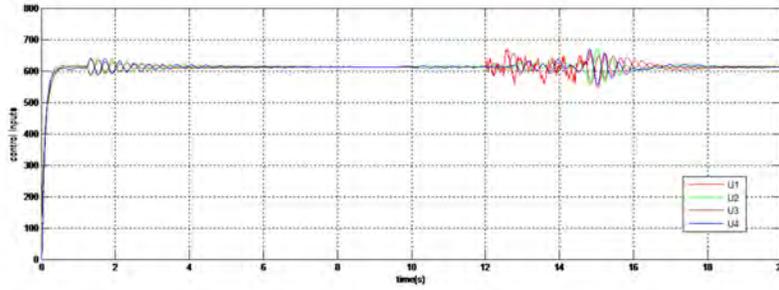
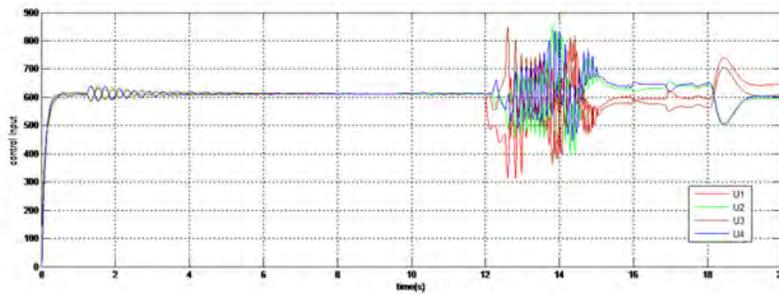


Figure 12: The tracking performance with actuator faults and noise



(a)



(b)

Figure 13: The control inputs of the quadrotor

quadrotor track the desired path accurately and smoothly. Even if an actuator loses part of lift force, the fault tolerant control can still ensure that the quadrotor can fly back to the steady value in a small deviation. In general, the fault tolerance control based on LADRC and TDC has good control performance.

5 Conclusion

In this paper, a fault tolerant control method based on LADRC and TDC is introduced to design the attitude and position controller of the quadrotor, which solves the robust and highly reliable flight issue of the quadrotor. The main contribution of this paper is that the design of the control method fully considers the system disturbances and actuator faults which occur frequently during the actual flight task. Based on this, the designed controller is more practical to actual flight. Moreover, the quadrotor model is linearized into three models, thus the fault tolerant control method is used in not only the attitude control but also the position control, which can make the control effect of the quadrotor more intuitively. Simulation studies are carried out comparing with PID method. The results show that the developed fault tolerant control method has a better control performance with or without actuator faults.

References

- [1] K. Alexis, C. Papachristos, G. Nikolakopoulos and A. Tzes, Model predictive quadrotor indoor position control, in: *Proceedings of 19th Mediterranean Conference on Control and Automation*, IEEE Computer Society, Greece, 2011, pp. 1247–1252.
- [2] A.F. Ali and M.A. Tawhid, Hybrid simulated annealing and pattern search method for solving minimax and integer programming problems, *Pacific. J. Optim.* 12 (2016) 151–184.
- [3] L. Besnard, Y. B. Shtessel and B. Landrum, Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer, *J. Franklin Inst.* 349 (2012) 658–684.
- [4] G. Caprari, A. Breitenmoser and W. Fischer, Highly compact robots for inspection of power plants, *J. Field Robot.* 29 (2012) 47–68.
- [5] P.H. Chang and J.W. Lee, A model reference observer for time-delay control and its application to robot trajectory control, *IEEE Trans. Control Syst. Technol.* 4 (1996) 2–10.
- [6] J. Cieslak, D. Henry and A. Zolghadri, Fault tolerant flight control: from theory to piloted flight simulator experiments, *IET Control Theory Appl.* 4 (2010) 1451–1464.
- [7] L. Derafa, T. Madani and A. Benallegue, Dynamic modelling and experimental identification of four rotors helicopter parameters, in: *Proceedings of the IEEE International Conference on Industrial Technology*, Institute of Electrical and Electronics Engineers Inc., India, 2006, pp. 1834–1839.
- [8] C. Fu, Y. Tian, C. Peng, X. Gong and Y. Bai, Path tracking control for eight-rotor aircraft based on linear ADRC algorithm, in: *Proceedings of the 2016 IEEE 11th Conference on Industrial Electronics and Applications*, Institute of Electrical and Electronics Engineers Inc., China, 2016, pp. 2147–2152.

- [9] Z.Q. Gao, From linear to nonlinear control means: a practical progression, *ISA trans.* 41 (2002) 177–189.
- [10] Z.Q. Gao, Scaling and bandwidth-parameterization based controller tuning, in: Proceedings of the American Control Conference, Institute of Electrical and Electronics Engineers Inc., USA, 2003, pp. 4989–4996.
- [11] S.R. Herwitz, L.F. Johnson, S.E. Dunagan, R.G. Higgins, D.V. Sullivan and J. Zheng, Imaging from an unmanned aerial vehicle: agricultural surveillance and decision support, *Comput. Electron. Agric.* 44 (2004) 49–61.
- [12] M. Huang, B. Xian, C. Diao, K. Yang and Y. Feng, Adaptive tracking control of underactuated quadrotor unmanned aerial vehicles via backstepping, in: *Proceeding of the 2010 American Control Conference*, IEEE Computer Society, USA, 2010, pp. 2076–2081.
- [13] J. Jin, S. Ko and C. K. Ryoo, Fault tolerant control for satellites with four reaction wheels, *Control Eng. Pract.* 16 (2008) 1250–1258.
- [14] E.N. Johnson, A.A. Proctor and J. Ha, Development and test of highly autonomous unmanned aerial vehicles, *J. Aerosp. Comput. Inf. Commun.* (2015) 578–586.
- [15] J. Kennedy and R. Eberhart, Particle swarm optimization, in: *Proceedings of International Conference on Neural Networks*, Institute of Electrical and Electronics Engineers Inc., Australia, 1995, pp. 1942–1948.
- [16] Z. Ma and S.M. Jiao, Research on the attitude control of quad-rotor UAV based on active disturbance rejection control, in: *Proceedings of International Conference on Control Science and Systems Engineering*, Institute of Electrical and Electronics Engineers Inc., China, 2017, pp. 45–49.
- [17] V. Martínez Martínez, *Modelling of the flight dynamics of a quadrotor helicopter*, A Msc Thesis in Cranfield University, 2007.
- [18] C. Peng, Y. Tian, Y. Bai, X. Gong and C. Zhao, ADRC trajectory tracking control based on PSO algorithm for a quadrotor, in: *Proceeding of the 8th IEEE Conference on Industrial Electronics and Applications*, IEEE Computer Society, Australia, 2013, pp. 800–805.
- [19] T. Plangsorn and C. Deelertpaiboon, Determination of object's coordinate using quadrotor for search and rescue mission, in: *Proceedings of IEEE Region 10 Annual International Conference*, Institute of Electrical and Electronics Engineers Inc., Thailand, 2015, pp.1–5.
- [20] P. Pounds, R. Mahony and P. Corke, Modelling and control of a large quadrotor robot, *Control Eng. Practice* 18 (2010) 691–699.
- [21] H.J. Qing, From pid technique to active disturbances rejection control technique, *Control Eng.* 9 (2010) 13–18.
- [22] B. Qiu, H. Xiong and J. Fu, The position control of micro quad-rotor UAV based on ADRC, in: *Proceedings of Chinese Automation Congress*, Institute of Electrical and Electronics Engineers Inc., China, 2016, pp. 422–426.

- [23] F. Sharifi, M. Mirzaei, B. W. Gordon and Y. Zhang, Fault tolerant control of a quadrotor UAV using sliding mode control, in: *Final Program and Book of Abstracts of Conference on Control and Fault-Tolerant Systems*, IEEE Computer Society, France, 2010, pp. 239–244.
- [24] D.D. Siljak, K.J. Astrom, G.F. Franklin, A.H. Levis and W.R. Perkins, Challenges to control: a collective view, *IEEE Trans. Automat. Control* AC-32 (1987) 275–585.
- [25] A. Wallar, E. Plaku and D. A. Sofge, Reactive motion planning for unmanned aerial surveillance of risk-sensitive areas, *IEEE Trans. on Autom. Sci. and Eng.* 12 (2015) 969–980.
- [26] H. Wang, X. Fang, Z. Cai and Y. Wang, Modeling and control of oblique cross quadrotors vehicle, in: *Proceedings of 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce*, IEEE Computer Society, China, 2011, pp. 3113–3117.
- [27] H. Wang, X. Fang, Z. Cai and Y. Wang, Modelling and control of oblique cross quadrotors vehicle, in: *Proceedings of 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce*, IEEE Computer Society, China, 2011, pp. 3113–3117.
- [28] L. Wang, C.J. Yu, E. M. Feng, Z.L. Xiu, P.J. Zhang and X.H. Chen, Optimal control of microbial fermentation in batch culture using particle swarm algorithm, *Pac. J. Optim.* 12 (2016) 437–449.
- [29] L. Xu, T. Cai and F. Deng, Sensor fault diagnosis based on least squares support vector machine online prediction, in: *Proceedings of the 2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics*, IEEE Computer Society, China, 2011, pp. 275–279.
- [30] Q. Yang, G. Bao and H. Zhang, Research of auto-disturbance-rejection controller in motor control system , *Int. J. Control Autom.* 7 (2014) 271–282.
- [31] Y.Q. Yang and G.B. Xiang, The new standard forms of ITAE optimum transfer function, *Information and Control* 26 (1997) 259–265.
- [32] K. Youcef-Toumi and S. Reddy, Analysis of linear time invariant systems with time delay, *J Dyn Syst Meas Control Trans ASME* 114 (1992) 544–555 .
- [33] J.R. Zhang, J. Zhang, T.M. Lok and M.R. Lyu, A hybrid particle swarm optimization back-propagation algorithm for feedforward neural network training, *Appl. Math. Comput.* 185 (2007) 1026–1037.

Manuscript received 5 December 2018
revised 2 April 2019
accepted for publication 5 May 2019

JIAJIN LI

School of Automation Engineering
University of Electronic Science and Technology of China
Chengdu, 611731, P.R. China
E-mail address: lijiajinsina@163.com

RUI LI

School of Automation Engineering
University of Electronic Science and Technology of China
Chengdu, 611731, P.R. China
E-mail address: lirui@uestc.edu.cn

YINGJING SHI

School of Automation Engineering
University of Electronic Science and Technology of China
Chengdu, 611731, P.R. China
E-mail address: shiyingjing@uestc.edu.cn

LIANG HE

Shanghai Key Laboratory of Aerospace Intelligent Control Technology
Shanghai, 201109, P.R. China
E-mail address: aerospace@vip.sina.com