International Journal of Innovative Computing, Information and Control
Volume **17**, Number **1**, February **2021**

ICIC International ©2021 ISSN 1349-4198
pp. **49–65**

SLIDING MODE PREDICTION FAULT-TOLERANT CONTROL OF A QUAD-ROTOR SYSTEM WITH MULTI-DELAYS BASED ON ICOA

Zhiqing Zhang, Pu Yang, Xukai Hu and Zixin Wang

College of Automation
Nanjing University of Aeronautics and Astronautics
No. 29, Jiangjun Avenue, Jiangning District, Nanjing 210016, P. R. China
ppyang@nuaa.edu.cn

Received August 2020; revised December 2020

Abstract. *A novel sliding mode prediction fault-tolerant control method is designed for a class of discrete uncertain quad-rotor systems with multi-delays and actuator faults in this paper. Firstly, a quasi-integral sliding mode surface is designed as a prediction model to eliminate the sliding mode approaching mode and ensure the global robustness. Secondly, aiming at actuator faults and multiple time delays, a double-power function reference trajectory with enhanced fault compensation is designed to reduce the impact of time delays on the system, and improve fault-tolerant control accuracy. Thirdly, an inverse time coyote optimization algorithm (ICOA) is designed for rolling optimization. Based on gaining good convergence speed, the ICOA can avoid local extremes and balance local development capabilities and global search capabilities. Finally, the comparison simulation study on the quad-rotor proves the superiority of the proposed control algorithm.*

Keywords: Quad-rotor, Sliding mode prediction, Multi-delays, Actuator faults, ICOA

1. Introduction.

Currently, the quad-rotor unmanned aerial vehicle (UAV) has received extensive attention from researchers due to its lightweight, convenient operation, low cost, and excellent environmental adaptability [1]. In real life, the quad-rotor helicopters have been widely applying in many aspects such as tracking and tracking obstacle avoidance [2,3], aerial photography [4,5], and formation flying [6,7]. However, there are plenty of hidden dangers in the actual application of quad-rotor, such as frequent failures and external disturbances [8]. Therefore, it becomes particularly critical to design excellent fault-tolerant control strategies for the security and stability of their control systems [9].

In recent years, domestic and foreign scientific researchers have conducted extensive research on the fault-tolerant control of quad-rotors with actuator failures and proposed a series of fault-tolerant control methods. They combined algorithms such as sliding mode control, predictive control, neural networks, and adaptive theory [10-13], to design more superior algorithms such as sliding mode prediction, and adaptive sliding mode [14,15]. [16] proposes a robust fault-tolerant formation control method for a class of tail-sitter unmanned aerial vehicle systems with actuator failures, coupling, and parameter uncertainties. [17] proposes a fixed-time controller and a robust adaptive controller based on integral sliding mode for the attitude stability of quad-rotor aircraft. However, this paper has not considered time delays, and it is challenging to implement output feedback control. In [18], for a class of uncertain discrete systems with actuator failures and state time-varying delays, a discrete predictive sliding mode fault-tolerant control method based

DOI: 10.24507/ijicic.17.01.49

on multi-agent particle swarm optimization is proposed. Since this paper adopts the traditional linear sliding mode surface, the instability is natural to occur during the sliding mode approach. Of course, scholars have made significant achievements not only in the field of passive fault-tolerant control algorithms, but in the area of active fault-tolerant control as well. [19] designs an active fault-tolerant control method based on adaptive sliding mode and recurrent neural network for the quad-rotor system with actuator failure and uncertainty. However, this method does not consider the influence of external interference and time lag. In [20], the authors propose an active fault-tolerant control strategy based on model predictive control (MPC), which first designs a second-order discrete sliding mode observer to observe the actuator fault information, and then uses it to reconstruct the control law to stabilize the system. However, the optimization algorithm adopts the method of directly seeking the extreme value, which makes the calculation redundant.

Nevertheless, the problems considered in the above literature mainly focus on the fault-tolerant control with actuator failure. In practical engineering applications, the state delay and input delay of the control system usually make the system more complex and sensitive. Therefore, its stability analysis and control have been widely concerned by scholars all over the world. In [21], in allusion to a kind of aero-engine bivariate control system with both input time delay and state time delay, the optimal sliding mode predictive control method based on time-delay compensator is studied. This paper first uses a linear transformation to transform the time-delay system into a system without distinct time-delay terms, and then uses a time-delay compensator to realize the advance control of the system. [22] proposes a global sliding mode control method based on LMI for a class of uncertain discrete descriptor systems with multiple time-varying delays. In [23], for a type of discrete memristive neural network system with multi-delays, a dynamic delay interval method is used to deal with the input-to-state stability problem. [24] proposes a method for the mean square stabilization of a discrete-time stochastic system with input time delay and multiplicative noise based on a discrete fuzzy controller. However, this article only considers the input time delay, and needs to ensure the stability of the general stochastic time-delay system.

Compared with the algorithm in [25], the sliding mode prediction algorithm designed in this paper further elaborates its superiority. This paper mainly studies the discrete uncertain system with actuator failure, multi-delays, and external disturbances. The quasi-integral sliding surface is designed as a prediction model to eliminate the system state's approaching process and ensure good robustness. A double power function reference trajectory is designed to effectively reduce the impact of time delays on the system. In the part of reference trajectory design, an improved fault compensation term is added to decrease the quasi-sliding mode bandwidth and improve the control accuracy. In the rolling optimization process, an improved inverse time coyote optimization algorithm (ICOA) is designed to accelerate algorithm convergence speed. Compared with the traditional coyote optimization algorithm (COA), the inverse time decay weight factor is introduced to maintain the balance of local development capabilities and global search capabilities.

The outline of the full text is as follows. The second part describes the system model and explains the relevant basic knowledge and assumptions. The third part designs the control algorithm of this paper. Firstly, based on an integral sliding mode surface, a sliding mode prediction model is designed in this paper to ensure global robustness. Secondly, a reference trajectory of the double power function is designed. Thirdly, an improved fault compensation term is added to the reference trajectory. Finally, an improved inverse time coyote optimization algorithm is developed. The fourth part is the stability proof. The fifth part is a simulation comparison test. It further proves the feasibility and superiority of the innovative algorithm designed in the third part. Finally, the last part summarizes the paper.

2. **System Model Analysis.** This article selects the Qball-X4 quad-rotor from Quanser Company in Canada. In Qball-X4 quad-rotor, the system has six dimensions of variables, that is $(X, Y, Z, \psi, \theta, \phi)$, where X, Y, Z is the position variable, ψ is the yaw angle, θ is the pitch angle and ϕ is the roll angle. This paper chooses the X-axis or Y-axis forward direction channel signal as the research object. Since the movement of the X-axis and Y-axis directions are symmetrical, only the X direction forward channel signal is considered below. The mathematical model is shown in Table 1 below.

Table 1. The mathematical model of Qball-X4

Physical interpretation	*Mathematical expression*
Dynamic equation of X axis	$M_g\ddot{X} = 4F\sin\left(\dot{\theta}\right)$
Lift generated by the rotor: F	$F = K_g\frac{\omega}{s+\omega}u$
Actuator dynamics: ν	$\nu = \frac{\omega}{s+\omega}u$
State expression form of the ν: $\dot{\nu}$	$\dot{\nu} = -\omega\nu + \omega u$
The model in the X-axis direction	$\begin{bmatrix} \dot{X} \\ \ddot{X} \\ \dot{\nu} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{4K_g}{M_g}\theta \\ 0 & 0 & -\omega \end{bmatrix} \begin{bmatrix} X \\ \dot{X} \\ \nu \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} u$

The above mathematical model assumes that the yaw angle is 0, and considers the influence of lift and pitch angle. Among them, M_g is the mass of the quad-rotor, F is the lift, K_g is the positive gain, ω is the actuator bandwidth, u is the actuator input, θ is the pitch angle; let $\sin\theta = \theta$. After processing, the model in the X-axis direction can be obtained. Discretize the above model first, and then consider the actuator failure, input time delay and state time delay, internal parameter perturbation and external disturbance, and the following discrete uncertain fault system is obtained:

$$\left\{\begin{aligned} & x(k+1) = (A + \Delta A)x(k) + (A_d + \Delta A_d)x(k-\tau_1) + (B + \Delta B)x \\ & \qquad\qquad + (B_d + \Delta B_d)u(k-\tau_2) + Df(k) + \upsilon(k) \\ & y(k) = Cx(k) \end{aligned}\right. \tag{1}$$

$x(k) \in R^n$, $u(k) \in R^p$, $y(k) \in R^q$, are the state, input and output of the system. A, B, A_d, B_d, C, D are the constant matrices. ΔA, ΔB, ΔA_d, ΔB_d are the internal perturbation of the system. $\tau_1, \tau_2 \in R^+$ are the state delay and input delay respectively, and they both have upper bounds $\tau_1 up$, $\tau_2 up$, $f(k)$ is an actuator fault, and $\upsilon(k) \in R^n$ is external disturbance.

System (1) can be rewritten as follows:

$$x(k+1) = Ax(k) + A_d x(k-\tau_1) + Bu(k) + B_d u(k-\tau_2) + d(k) \tag{2}$$

$d(k)$ describes the faults and uncertainties of the system.

$$d(k) = \Delta Ax(k) + \Delta A_d x(k-\tau_1) + \Delta Bu(k) \ + \Delta B_d u(k-\tau_2) + d_f(k) \tag{3}$$

$$d_f(k) = Df(k) + \upsilon(k) \tag{4}$$

So $\hat{d}(k)$ can be obtained by one-step delay estimation method,

$$\begin{aligned} \hat{d}(k) &= d(k-1) \\ &= x(k) - Ax(k-1) - A_d x(k-1-\tau_1) - Bu(k-1) - B_d u(k-1-\tau_2) \end{aligned} \tag{5}$$

We can get a simplified system as (6):

$$\begin{cases} x(k+1) = Ax(k) + A_d x(k-\tau_1) + Bu(k) + B_d u(k-\tau_2) + d(k) \\ y(k) = Cx(k) \end{cases} \tag{6}$$

Assumption 2.1. *The rate of change of the system's failure and uncertainty is bounded.*

$$|d(k) - d(k-1)| \le d_0$$

Assumption 2.2. *The failure and uncertainty of the system have an upper and lower bound.*

$$d_L \le |d(k)| \le d_U$$

Lemma 2.1. *(Schur's complement theorem) For a given symmetric matrix* $\begin{bmatrix} \Omega_{11} & \Omega_{12} \\ \Omega_{21} & \Omega_{22} \end{bmatrix} < 0$, *where* $\Omega_{11}^T = \Omega_{11}$, $\Omega_{22}^T = \Omega_{22}$, $\Omega_{12}^T = \Omega_{21}$. *Then the above equation is equivalent to* (1) $\Omega_{11} < 0$, $\Omega_{12}\Omega_{11}^{-1}\Omega_{12} < 0$*;* (2) $\Omega_{22} < 0$, $\Omega_{12}\Omega_{22}^{-1}\Omega_{21} < 0$.

3. Control Algorithm Design.

3.1. Sliding mode prediction model design.

The quasi-integral sliding mode switching function is designed in this section, making the initial state of the system on the sliding surface, which is different from the global sliding mode switching function in [25]. The quasi-integral sliding mode switching function makes the initial state of the system on the sliding surface, which eliminates the sliding mode approaching mode, making the system globally robust from the beginning.

$$\begin{cases} s(k) = Gx(k) + \sigma(k) - Gx(0) \\ \sigma(k+1) = \sigma(k) + Gx(k) - GAx(k) - GA_d x(k-\tau_1) \end{cases} \tag{7}$$

where $\sigma(0) = 0$, $G \in R^{p\times n}$ is a constant matrix that satisfies GB is nonsingular. We can get the following predicted output at the moment $(k+P)$ of the sliding mode prediction model according to (6):

$$\begin{aligned} s(k+P) = G\Bigg[& A^P x(k) + \sum_{i=1}^{P} A^{i-1} A_d x(k+P-i-\tau_1) \\ & + \sum_{i=1}^{M-1} A^{P-i} Bu(k+i-1) + \sum_{i=1}^{P-M} A^i Bu(k+M-1) \\ & + \sum_{i=1}^{M+\tau_2(k)-1} A^{P-i} B_d u(k+i-1-\tau_2) \\ & + \sum_{i=1}^{P-M-\tau_2(k)} A^i B_d u(k+M-1) \Bigg] + \sigma(k+P) - Gx(0) \end{aligned} \tag{8}$$

where P denotes the prediction time domain, and M denotes the control time domain. The vector form of (8) is

$$S_{PM}(k) = \Lambda X(k) + \Phi X_d(k) + \Pi U(k) + \mathrm{E} U_d(k) + \Sigma(k) \tag{9}$$

where

$$S_{PM}(k) = [s(k+1), s(k+2), \ldots, s(k+P)]^T;$$

$$\Lambda = \left[(GA)^T, \left(GA^2\right)^T, \ldots, \left(GA^P\right)^T \right]^T;$$

$$X(k) = [x(k+1), \ldots, x(k+P)]^T;$$

$$X_d(k) = [x(k-\tau_1(k)), x(k+1-\tau_1(k+1)), \dots, x(k+P-1-\tau_1(k+P-1))]^T;$$
$$U(k) = [u(k), u(k+1), \dots, u(k+M-1)]^T;$$
$$U_d(k) = [u(k-\tau_2(k)), u(k+1-\tau_2(k+1)), \dots, u(k+M-1)]^T;$$
$$\Sigma(k) = [\sigma(k+1) - Gx(0), \sigma(k+2) - Gx(0), \dots, \sigma(k+P) - Gx(0)]^T;$$

$$\mathrm{E} = \begin{bmatrix} GB_d & 0 & \cdots & \cdots & 0 \\ GAB_d & GB_d & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \cdots & \vdots \\ GA^{M-1}B_d & GA^{M-2}B_d & \cdots & GAB_d & GB_d \\ GA^M B_d & GA^{M-1}B_d & \cdots & GA^2B_d & GAB_d + GB_d \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ GA^{P-1}B_d & GA^{P-2}B_d & \cdots & GA^{P-M+1}B_d & \sum\limits_{i=0}^{P-M} GA^iB_d \end{bmatrix};$$

$$\Phi = \begin{bmatrix} GA_d & 0 & \cdots & \cdots & 0 \\ GAA_d & GA_d & \cdots & \cdots & 0 \\ \vdots & \vdots & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ GA^{P-1}A_d & GA^{P-2}A_d & \cdots & \cdots & GA_d \end{bmatrix};$$

$$\Pi = \begin{bmatrix} GB & 0 & \cdots & \cdots & 0 \\ GAB & GB & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \cdots & \vdots \\ GA^{M-1}B & GA^{M-2}B & \cdots & GAB & GB \\ GA^M B & GA^{M-1}B & \cdots & GA^2B & GAB + GB \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ GA^{P-1}B & GA^{P-2}B & \cdots & GA^{P-M+1}B & \sum\limits_{i=0}^{P-M} GA^iB \end{bmatrix}.$$

3.2. **Prediction model stability analysis.** After establishing the quasi-integral sliding mode prediction model, its stability is crucial to the construction of subsequent algorithms. The stability analysis of the prediction model is demonstrated below. We can get equivalent control law according to $s(k+1) = s(k) = 0$, and then we can get

$$\begin{aligned} u_{eq}(k) &= -(GB)^{-1}Gd(k) \\ &= -(GB)^{-1}G\left[\Delta Ax(k) + \Delta A_d x(k-\tau_1) + \Delta Bu(k)\right. \\ &\quad \left. + \Delta B_d u(k-\tau_2) + Df(k) + \upsilon(k)\right] \end{aligned} \tag{10}$$

Substituting (10) to the system (6), the ideal sliding mode equation can be obtained as

$$x(k+1) = Ax(k) + A_d x(k-\tau_1) + B_d u(k-\tau_2) + \left[I - B(GB)^{-1}G\right] d(k) \tag{11}$$

Theorem 3.1. *For system (6), the quasi-integral sliding mode prediction model determined by (7), if there is a positive definite matrix Q_i $(i = 1, 2, 3)$, satisfying Inequality (12), the ideal sliding mode Equation (11) is globally asymptotically stable.*

$$\begin{bmatrix} \Psi_1 & 0 & 0 & 0 & 0 \\ * & \Psi_2 & 0 & 0 & 0 \\ * & * & \Psi_3 & 0 & 0 \\ * & * & * & 4Q_1 & \sqrt{2}Q_1B \\ * & * & * & * & -B^TQ_1B \end{bmatrix} < 0 \tag{12}$$

where $\Psi_1 = 5A^TQ_1A - Q_1 + Q_2$, $\Psi_2 = 4{A_d}^TQ_1A_d + {A_d}^TQ_2A_d - Q_2$, $\Psi_3 = 4{B_d}^TQ_1B_d + {A_d}^TQ_2A_d - Q_3$.

Proof: Choose the following Lyapunov function for Equation (11):

$$V(k) = x^T(k)Q_1x(k) + \sum_{i=k-\tau_1(k)}^{k-1} x^T(i)Q_2x(i) + \sum_{j=k-\tau_2(k)}^{k-1} u^T(j)Q_3u(j) \tag{13}$$

Choose $G = B^TQ_1$ to ensure GB is nonsingular, and make $R = \left(B^TQ_1B\right)^{-1} B^TQ_1$.

The difference equation of the Lyapunov function along the state trajectory of the ideal sliding mode (11) satisfies:

$$\begin{aligned}
\Delta V(k) &= V(k+1) - V(k)\\
&= x^T(k+1)Q_1x(k+1) + \sum_{i=k+1-\tau_1(k+1)}^{k} x^T(i)Q_2x(i) + \sum_{j=k+1-\tau_2(k+1)}^{k} u^T(j)Q_3u(j)\\
&\quad - x^T(k)Q_1x(k) - \sum_{i=k-\tau_1(k)}^{k-1} x^T(i)Q_2x(i) - \sum_{j=k-\tau_2(k)}^{k-1} u^T(j)Q_3u(j)\\
&= x^T(k)A^TQ_1Ax(k) + x^T(k)(Q_2 - Q_1)x(k) + 2x^T(k)A^TQ_1A_dx(k-\tau_1)\\
&\quad + 2x^T(k)A^TQ_1B_du(k-\tau_2) + x^T(k-\tau_1){A_d}^TQ_1A_dx(k-\tau_1)\\
&\quad - x^T(k-\tau_1)Q_2x(k-\tau_1) + 2x^T(k-\tau_1){A_d}^TQ_2B_du(k-\tau_2)\\
&\quad + u^T(k-\tau_2){B_d}^TQ_1B_du(k-\tau_2) - u^T(k-\tau_2)Q_3u(k-\tau_2)\\
&\quad + 2x^T(k)A^TQ_1d(k) - 2x^T(k)A^TQ_1BRd(k) + 2x^T(k-\tau_1){A_d}^TQ_1d(k)\\
&\quad - 2x^T(k-\tau_1){A_d}^TQ_1BRd(k) + 2u^T(k-\tau_2){B_d}^TQ_1d(k)\\
&\quad - 2u^T(k-\tau_2){B_d}^TQ_1BRd(k) + d^T(k)Q_1d(k) - d^T(k)Q_1BRd(k)\\
&\le 5x^T(k)A^TQ_1Ax(k) + x^T(k)(Q_2 - Q_1)x(k) + 4x^T(k-\tau_1){A_d}^TQ_1A_dx(k-\tau_1)\\
&\quad + x^T(k-\tau_1){A_d}^TQ_2A_dx(k-\tau_1) - x^T(k-\tau_2)Q_2x(k-\tau_1)\\
&\quad + 4u^T(k-\tau_2){B_d}^TQ_1B_du(k-\tau_2) + u^T(k-\tau_2){B_d}^TH_2B_d\bar{u}(k-\tau_2)\\
&\quad - u^T(k-\tau_2)Q_3u(k-\tau_2) + 4d^T(k)Q_1d(k) + 2d^T(k)Q_1BRd(k)\\
&= \begin{bmatrix} x(k) & x(k-\tau_1) & u(k-\tau_2) & d(k) \end{bmatrix} \Lambda \begin{bmatrix} x(k)\\ x(k-\tau_1)\\ u(k-\tau_2)\\ d(k) \end{bmatrix}^T
\end{aligned} \tag{14}$$

where $\Lambda = \begin{bmatrix} \Psi_1 & & & \\ & \Psi_2 & & \\ & & \Psi_3 & \\ & & & \Psi_4 \end{bmatrix}$, $\Psi_4 = 4Q_1 + 2Q_1BR$. We can get $\Delta V(k) < 0$ according to $\Lambda < 0$. According to Lemma 2.1, $\Lambda < 0$ is equivalent to (12). Therefore, when the linear matrix inequality (12) is established, the ideal sliding mode (11) is asymptotically stable. The proof is complete.

3.3. **Reference trajectory design.** This section adopts the following reference trajectory based on the improved fault compensation double power function. The system still has strong robustness outside the boundary layer through the traditional "boundary layer" chattering suppression strategy. It also reduces the quasi-sliding mode bandwidth as far as possible in the boundary layer to decrease the steady-state error. Firstly, the difference function $1 - z^{-1}$ (where z^{-1} is the unit delay operator) is introduced. The equivalent change rate of faliure is defined as the second-order difference of the fault, which can reduce the quasi-sliding mode bandwidth. Secondly, compared to [25], this paper uses a double power function to replace the traditional power function, which can weaken the chattering while ensuring its robustness. The dynamic convergence speed can be accelerated, and the control precision of the system can be improved.

Reference trajectory in this paper:

$$\begin{cases} s_{ref}(k+1) = (1-qT)s_{ref}(k) - \varepsilon_1 T|s_{ref}(k)|^{\alpha} - \varepsilon_2 T|s_{ref}(k)|^{\beta}\mathrm{sgn}(s_{ref}(k)) \\ \qquad\qquad + (1-z^{-1})G[d(k) - d(k-1)] \\ s_{ref}(k) = s(k) \end{cases} \tag{15}$$

where $\varepsilon_1 > 0$, $\varepsilon_2 > 0$, $q > 0, 1 - qT > 0$, $0 < \alpha < 1$, $\beta > 1$, T is sampling time.

After simplification, the final form of the following reference trajectory can be obtained as follows:

$$\begin{cases} s_{ref}(k+1) = (1-qT)s_{ref}(k) - \varepsilon_1 T|s_{ref}(k)|^{\alpha} \\ \qquad\qquad - \varepsilon_2 T|s_{ref}(k)|^{\beta}\mathrm{sgn}(s_{ref}(k)) + \lambda(k) \\ s_{ref}(k) = s(k) \end{cases} \tag{16}$$

where

$$\lambda(k) = G[d(k) - 2d(k-1) + d(k-2)] \tag{17}$$

Lemma 3.1. *From [26], when the zero-order holder is used for discretization, the equivalent failure in the system (6) $d(k)$ has properties as follows:*

$$d(k) - 2d(k-1) + d(k-2) = O\left(T^3\right)$$

where $O(T)$ indicates that the magnitude of $d(k)$ is on the order of magnitude $O(T)$.

Assumption 3.1. *The change rate of equivalent failure $\lambda(k)$ defined by (17) is bounded, and $|\lambda(k)| \le \delta \le \varepsilon T$, δ is the upper bound of $\lambda(k)$.*

Remark 3.1. *According to [27], the traditional change rate of equivalent failure $\lambda_1(k)$ and its upper bound δ_1 is defined as:*

$$\begin{cases} \lambda_1(k) = G[d(k) - d(k-1)] \\ |\lambda_1(k)| \le \delta_1 \end{cases} \tag{18}$$

3.4. **Feedback correction.** Define $s(k|k-P)$ as the predicted output of time $(k-P)$ to time k, we can get the following:

$$\begin{aligned} s(k\,|k-P) = G\Bigg[&A^P x(k-P) + \sum_{i=1}^{P} A^{i-1}A_d x(k-i-\tau_1) \\ &+ \sum_{i=1}^{M-1} A^{P-i}Bu(k-P+i-1) + \sum_{i=1}^{P-M} A^i Bu(k-P+M-1) \\ &+ \sum_{i=1}^{M+\tau_2(k)-1} A^{P-i}B_d u(k-P+i-1-\tau_2) \\ &+ \sum_{i=1}^{P-M-\tau_2(k)} A^i B_d u(k-P+M-1)\Bigg] + \sigma(k) - Gx(0) \end{aligned} \tag{19}$$

$e_s(k)$ is the error of predicted output and actual output at the time k:

$$e_s(k) = s(k) - s(k|k-P) \tag{20}$$

Then after error correction and P step prediction, the output machine vector form is as follows:

$$\widetilde{s}(k+P) = s(k+P) + f_p e_s(k) \tag{21}$$

$$\tilde{S}_{PM}(k) = S_{PM}(k) + F_P E_S(k) \tag{22}$$

where

$$\tilde{S}_{PM}(k) = [\widetilde{s}(k+1), \widetilde{s}(k+2), \ldots, \widetilde{s}(k+P)]^T;$$

$$F_P = \begin{bmatrix} f_1 & & & \\ & f_2 & & \\ & & \ddots & \\ & & & f_P \end{bmatrix},\ 1 \geq f_1 \geq f_2 \geq \cdots \geq f_P > 0;$$

$$E_S(k) = [s(k) - s(k|k-1), s(k) - s(k|k-2), \ldots, s(k) - s(k|k-P)]^T.$$

3.5. **Optimization algorithm design.** The optimized performance index and its vector form at time k are as follows:

$$j(k) = \sum_{i=1}^{P} \lambda_i [s_{ref}(k+i) - \tilde{s}(k+1)]^2 + \sum_{l=1}^{M} \rho_l [u(k+l-1)]^2 \tag{23}$$

$$J(k) = \left[S_{ref}(k) - \tilde{S}_{PM}(k)\right]^T Q_5 \left[S_{ref}(k) - \tilde{S}_{PM}(k)\right] + [U(k)]^T Q_4 [U(k)] \tag{24}$$

λ_i, ρ_l are non-negative weight, $Q_4 = \begin{bmatrix} \rho_1 & & & \\ & \rho_2 & & \\ & & \ddots & \\ & & & \rho_M \end{bmatrix}$, $Q_5 = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_P \end{bmatrix}$.

COA is a unique intelligent bionic optimization algorithm with a short time to propose. The COA simulates the birth, growth, death, and migration of the coyote population. It has a better search model, framework, and has strong local and global search capabilities. The COA makes the individuals group randomly after initialization, and the coyotes in the group are randomly expelled and accepted, which enables the exchange of information between groups. However, when solving some high-dimensional complex functions, it is easy to fall into the local optimum, and is challenging to jump out of the local extremum. Besides, the algorithm has always maintained a constant update mechanism, which weakens the algorithm's global searchability.

In response to the above problems, this article adopts ICOA. The inverse time decay weight factor is introduced into the updated formula of coyote individual to maintain the balance between local development capabilities and global search capabilities, while speeding up the algorithm convergence speed.

1) Set the parameters: the coyote group N_p, the number of coyote individuals in each group N_c, the dimension D, and the termination condition nfevalMAX, and so on. Randomly initialize the coyote group. The i-th coyote individual in the p group at time t is defined as:

$$y_w^{p,t} = lb_w + r_w(ub_w - lb_w) \tag{25}$$

$$y_i^{p,t} = \left(y_1^{p,t}, y_2^{p,t}, y_3^{p,t}, \ldots, y_{N_c}^{p,t}\right) \tag{26}$$

where ub_w, lb_w respectively represent the upper and lower bounds of the w-th dimension, and r_w is a randomly generated real number in the range $[0, 1]$.

2) Evaluate the adaptability of coyotes.

$$Adapt_i^{p,t} = A\left(y_i^{p,t}\right) \tag{27}$$

The probability of coyote population change is P_e:

$$P_e = 0.005N_c^2 \tag{28}$$

3) The head wolf in the current group is $Cbest^{p,t}$, the current cultural trend of coyotes $clut^{p,t}$:

$$Cbest^{p,t} = \left\{ y_i^{p,t} \left\| \arg_{\{i=1,2,\ldots,N_c\}} \min Adapt\left(y_i^{p,t}\right) \right. \right\} \tag{29}$$

$$clut_j^{p,t} = \begin{cases} O^{p,t}_{\frac{N_c+1}{2},w}, & N_c \text{ is odd} \\ O_{\frac{N_c}{2},w} + O^{p,t}_{\frac{N_c+1}{2},w}, & \text{others} \end{cases} \tag{30}$$

where $O^{p,t}_{\frac{N_c+1}{2},w}$ indicates the median of the w-th dimension variable of all coyotes in p group at time t, when N_c is odd.

4) Birth and death of coyotes: record the age of the coyote (in years) as $year_i^{p,t}$. The birth of a new coyote is defined as ($pup^{p,t}$):

$$pup_w^{p,t} = \begin{cases} y^{p,t}_{n1,w}, & rnd_w < P_s \text{ or, } w = w_1 \\ y^{p,t}_{n2,w}, & rnd_w \geq P_s + P_a \text{ or, } w = w_2 \\ R_w, & rnd_w, \text{ else} \end{cases} \tag{31}$$

where n_1, n_2 are arbitray coyotes from the group p. w_1, w_2 are the two random dimensions of the problem. R_w, rnd_w are random numbers generated in $[0, 1]$ by uniform probability. (P_s) is discrete probability, (P_a) is associated probability, and they influence the cultural diversity of individuals in the coyote pack:

$$P_s = \frac{1}{D},\ P_a = \frac{(1 - P_s)}{2} \tag{32}$$

ω represents that the adaptability of coyotes in the group is not as good as that of the cubs, and ϕ represents the number of coyotes in the current group. If ϕ is 1 and ω is established, it means that there is one coyote in the group, and the adaptability of young wolves is better than that of the only coyote. Then the young wolf survives and the only coyote in the group dies. If ϕ is greater than 1 and ω is established, then the young wolf survives and the oldest coyotes in the group die. Otherwise, the young wolf dies.

5) Calculate the influence of head coyote and group cultural trends on the individual update in the coyote group at the current moment, δ_1, δ_2:

$$\delta_1 = Cbest^{p,t} - y^{p,t}_{cr_1},\ \delta_2 = cult^{p,t} - y^{p,t}_{cr_2} \tag{33}$$

where cr_1, cr_2 respectively represent the random coyote in the current group.

6) Update all the coyote individuals in the coyote group in turn to obtain new coyote individuals $new\_y_i^{p,t}$, choose the best adaptability between the new coyote and the original coyote, and keep the best coyote $y_i^{p,t+1}$:

$$new\_y_i^{p,t} = y_i^{p,t} + \kappa_1\delta_1 + \kappa_2\delta_2 \tag{34}$$

$$y_i^{p,t+1} = \begin{cases} new\_y_i^{p,t}, & f\left(new\_y_i^{p,t}\right) < f\left(y_i^{p,t}\right) \\ y_i^{p,t}, & \text{others} \end{cases} \tag{35}$$

Introduce the inverse time decay inertia weighting factor ω to the coyote individual update formula:

$$new\_y_i^{p,t} = \omega y_i^{p,t} + \kappa_1\delta_1 + \kappa_2\delta_2 \tag{36}$$

ω is the inertial weight of the coyote social state:

$$\omega(k) = (1 + gamma \cdot k)^{-p},\ gamma = 0.99 \tag{37}$$

where k is the current iteration number. To prevent the loss of the optimal solution that may be caused by the excessive decay rate, we define $p = 0.25$. κ_1, κ_2 are random real numbers in the range $[0,1]$, and they represent the weight of the individual coyote affected by the *alpha* coyote and group cultural trends.

7) Simulate the growth of the individual over time, and update the age of the individual coyote.

8) Judging the termination condition, if reached, output the social state of the coyote with the best adaptability; otherwise, return to step 2) to continue.

4. **Stability Analysis.** Aiming at system (6), making k moment the current moment, the predicted output at $k+P$ moment is

$$\begin{aligned} s(k+P) = G\Bigg[& A^P x(k) + \sum_{i=1}^{P} A^{i-1} A_d x(k+P-i-\tau_1) \\ & + \sum_{i=1}^{M-1} A^{P-i} B u(k+i-1) + \sum_{i=1}^{P-M} A^i B u(k+M-1) \\ & + \sum_{i=1}^{M+\tau_2(k)-1} A^{P-i} B_d u(k+i-1-\tau_2) \\ & + \sum_{i=1}^{P-M-\tau_2(k)} A^i B_d u(k+M-1) + \sum_{i=1}^{P} A^{i-1} d(k+P-1)\Bigg] \\ & + \sigma(k+P) - Gx(0) \end{aligned} \tag{38}$$

The actual predicted output vector form is as follows:

$$S_{PM}(k) = \Lambda X(k) + \Pi U(k) + \Phi X_d(k) + \mathrm{E} U_d(k) + \Theta\Omega(k) + \Sigma(k) \tag{39}$$

where $\Theta = \begin{bmatrix} G & 0 & \cdots & 0 \\ GA & G & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ GA^{P-1} & GA^{P-2} & \cdots & G \end{bmatrix}$, $\Omega(k) = \begin{bmatrix} d(k) & d(k+1) & \cdots & d(k+P-1) \end{bmatrix}^T$.

$\frac{\partial J(k)}{\partial U(k)} = 0$ is a necessary condition for $J(k)$ to take the extreme value, so the optimal control law must satisfy $\frac{\partial J(k)}{\partial U(k)} = 0$:

$$\begin{aligned} U(k) = {} & \left(Q_4 + \Pi^T Q_5 \Pi\right)^{-1} \Pi^T Q_5 [S_{ref}(k) - \Lambda X(k) - \Phi X_d(k) \\ & - \mathrm{E} U_d(k) - \Sigma(k) + F_P E_S(k)] \end{aligned} \tag{40}$$

Substitute (40) into (39):

$$\begin{aligned} S_{PM}(k) = {} & \Lambda X(k) + \Phi X_d(k) + \mathrm{E} U_d(k) + \Theta\Omega(k) + \Sigma(k) \\ & + \Pi\Big[\left(Q_4 + \Pi^T Q_5 \Pi\right)^{-1} \Pi^T Q_5 \left[S_{ref}(k) - \Lambda X(k)\right. \\ & \left. - \Phi X_d(k) - \mathrm{E} U_d(k) - \Sigma(k) + F_P E_S(k)\right]\Big] \end{aligned} \tag{41}$$

Considering robust stability, we usually take $Q_4 = 0$.

$$S_{PM}(k) = S_{ref}(k) + \Theta\Omega(k) + F_P E_S(k) \tag{42}$$

In the process of solving the control law by rolling optimization, only the current control input signal is implemented in the controlled object.

$$\begin{aligned} s(k+1) &= \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} S_{PM}(k) \\ &= s_{ref}(k+1) + f_1[s(k) - s(k \mid k-1)] + Gd(k) \\ &= s_{ref}(k+1) + G[d(k) - f_1 d(k-1)] \end{aligned} \tag{43}$$

Taking $f_1 = 1$, then (43) is simplified as follows:

$$s(k+1) = s_{ref}(k+1) + G[d(k) - d(k-1)] \tag{44}$$

According to Assumption 2.1, $|d(k) - d(k-1)| \le d_0$, we can get

$$s(k+1) = s_{ref}(k+1) + G[d(k) - d(k-1)] \le s_{ref}(k+1) + Gd_0 \tag{45}$$

where

$$s_{ref}(k+1) = (1 - qT)s_{ref}(k) - \varepsilon_1 T|s_{ref}(k)|^{\alpha} - \varepsilon_2 T|s_{ref}(k)|^{\beta}\mathrm{sgn}\left(s_{ref}(k)\right) + \lambda(k) \tag{46}$$

According to Assumption 3.1, $|\lambda(k)| \le \delta$, therefore, we only need to discuss the boundedness of the double-power exponential reaching law, that is, to judge the following formula:

$$s_{\varpi} = (1 - qT)s_{ref}(k) - \varepsilon_1 T|s_{ref}(k)|^{\alpha} - \varepsilon_2 T|s_{ref}(k)|^{\beta}\mathrm{sgn}\left(s_{ref}(k)\right) \tag{47}$$

Let $\Delta s_{\varpi}(k) = s_{\varpi}(k+1) - s_{\varpi}(k)$, and then we get

$$\Delta s_{\varpi}(k) = -qTs_{\varpi}(k) - \varepsilon_1 T|s_{\varpi}(k)|^{\alpha}\mathrm{sgn}(s(k)) - \varepsilon_2 T|s_{\varpi}(k)|^{\beta}\mathrm{sgn}(s(k)) \tag{48}$$

1) When $s_{\varpi}(k) \ge 0$:
$\frac{\partial \Delta s_{\varpi}(k)}{\partial s_{\varpi}(k)} = -qT - \varepsilon_1 T\alpha[s_{\varpi}(k)]^{\alpha-1} - \varepsilon_2 T\beta[s_{\varpi}(k)]^{\beta-1}$

Then according to $\varepsilon_1 > 0$, $\varepsilon_2 > 0$, $q > 0$, $1 - qT > 0$, $0 < \alpha < 1$, $\beta > 1$, we can obtain $\frac{\partial \Delta s_{\varpi}(k)}{\partial s_{\varpi}(k)} < 0$, and get $\Delta s_{\varpi}(k)$ is a decreasing function of $s_{\varpi}(k)$.

When $s_{\varpi}(k) \ge 0$, $\Delta s_{\varpi}(k) \le -qTs_{\varpi}(k) - \varepsilon_1 T[s_{\varpi}(k)]^{\alpha} - \varepsilon_2 T[s_{\varpi}(k)]^{\beta}\Big|_{s_{\varpi}(k)=0} = 0$. From $\Delta s_{\varpi}(k) \le 0$, we can get, $s_{\varpi}(k)$ decrease until approaching state $s_{\varpi}(k) = 0$. If and only if $s_{\varpi}(k) = 0$, $\Delta s_{\varpi}(k) = 0$, then $s_{\varpi}(k+1) \le \sigma$.

2) When $s_{\varpi}(k) < 0$:
$\frac{\partial \Delta s_{\varpi}(k)}{\partial s_{\varpi}(k)} = -qT - \varepsilon_1 T\alpha[-s_{\varpi}(k)]^{\alpha-1} - \varepsilon_2 T\beta[-s_{\varpi}(k)]^{\beta-1}$

And according to $\varepsilon_1 > 0$, $\varepsilon_2 > 0$, $q > 0$, $1 - qT > 0$, $0 < \alpha < 1$, $\beta > 1$, we can obtain $\frac{\partial \Delta s_{\varpi}(k)}{\partial s_{\varpi}(k)} < 0$, that is, $\Delta s_{\varpi}(k)$ is a decreasing function of $s_{\varpi}(k)$.

$\Delta s_{\varpi}(k) > -qTs_{\varpi}(k) - \varepsilon_1 T[s_{\varpi}(k)]^{\alpha} - \varepsilon_2 T[s_{\varpi}(k)]^{\beta}\Big|_{s_{\varpi}(k)=0} = 0.$

According to $\Delta s_{\varpi}(k) > 0$, we can get $s_{\varpi}(k)$ increase until approaching state $s_{\varpi}(k) = 0$. Then, $s_{\varpi}(k+1) \le \sigma$.

3) When $s_{\varpi}(k) = 0$:

$s_{\varpi}(k+1) = s_{\varpi}(k) = 0$, the system enters a stable state. Then, we can obtain $s_{\varpi}(k+1) \le \sigma$. In summary, we can get $s_{\varpi}(k+1) \le \sigma$, and since $s(k+1) \le s_{\varpi}(k+1) + \delta + Gd_0$, then, we can obtain $|s(k+1)| \le \sigma + \delta + Gd_0$. That is, the closed-loop system is robust stable.

5. **Simulation Experiments and Discussions.**

5.1. **Simulation model.**

The simulation object selected in this paper is the Qball-X4 quad-rotor. Then, the feasibility and effectiveness of the algorithm designed in this paper are verified on this aircraft.

Considering external disturbance, internal parameter perturbation, time delays, and actuator failure, each matrix's values in the quad-rotor system are as follows: $A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 12 \\ 0 & 0 & -15 \end{bmatrix}$, $A_d = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 4 \\ 0 & 0 & -5 \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ 0 \\ 15 \end{bmatrix}$, $B_d = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$, $C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$. Constant matrix $D = \begin{bmatrix} 0.2 & 0.4 & 0.1 \end{bmatrix}\sin(k)$. Uncertain parameters are $\Delta A = 0.1A$,

$\Delta A_d = 0.1A_d$, $\Delta B = 0.1B$, $\Delta B_d = 0.1B_d$, fault function $f(k) = \begin{bmatrix} 0.3\sin(6k) \\ 0.2\sin(3k) \\ 0.2\sin(2k) \end{bmatrix}$, external disturbance $w(k) = rand\sin^2(k)$.

The signal transmission of the quad-rotor system used in the simulation in this article is not transmitted through the wired network, but through wireless communication, it is prone to state time lag, so we set $\tau_1 = 5$. Since the control input may also bring time lag, the input time lag needs to be considered, and we set $\tau_2 = 3$. Sliding mode parameter matrix $\sigma = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$. We select the prediction time domain as $P = 4$, and take the control time domain as $M = 2$, the sampling time is taken as $k = 0.02$, and the simulation time domain is taken as $k = 500$. The above matrices in the numerical experiment are all discretized.

5.2. **Optimization algorithm implementation steps.** Pseudo-code of the ICOA algorithm is shown as follows:

```
START
Set parameters $N_c$, $N_p$; initialize coyote population (Equation (25))
Calculate the fitness of each coyote (Equation (27))
WHILE termination conditions are not met
$year = year + 1$
FOR $p = 1 : N_p$
Calculate head coyote and cultural trend (Equations (29)-(30))
Calculate the influence of the head coyote and groups (Equation (30))
FOR $i = 1 : N_c$
Update current coyote (Equations (36)-(38)), compare the adaptability of the coyotes before and after the update, and keep the best coyotes (Equation (35))
END $i$
Record births and deaths
END $p$
Group changes
IF Meet the threshold
END IF
Age update
END WHILE
Choosing the best coyote
END
```

In order to further test the performance advantages of the ICOA, this paper uses whale optimization algorithm (WOA), COA and ICOA to conduct comparative experiments on the same set of benchmark functions. Benchmark functions are shown in Table 2, where F_1 and F_2 are unimodal functions, F_3 and F_4 are multimodal functions.

In order to avoid the influence of randomness on the experimental results, and guarantee the absolute fairness of the test environment, in the simulation test, the common parameter settings of all algorithms are the same. We set the dimension D to 100, the maximum number of iterations to 50 in this paper, and $N_c = N_p = 10$. The termination condition of COA and ICOA is $nfevalMAX = 10000 * D$. WOA's parameter setting: logarithmic spiral shape constant $b = 1$. The number of search individuals is set to 10. Three algorithms were performed 30 times. The experimental results are shown in Table 3.

Table 2. Information of benchmark functions

Function form	*Variable range*
$F_1 = \sum\limits_{i=1}^{N} i{x_i}^2$	$[-100, 100]$
$F_2 = \sum\limits_{i=1}^{n} \lvert x_i \rvert + \Pi_{i=1}^{n} \lvert x_i \rvert$	$[-10, 10]$
$F_3 = \sum\limits_{i=1}^{n} \frac{{x_i}^2}{4000} - \Pi_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]$
$F_4 = -20 \exp\left(-0.2\sqrt{\frac{1}{n}\sum\limits_{i=1}^{n} {x_i}^2}\right) - \exp\left(\frac{1}{n}\sum\limits_{i=1}^{n} \cos(2\pi x_i)\right) + 20 + e$	$[-32, 32]$

Table 3. Comparison on experimental results based on WOA, COA and ICOA

Function	*Evaluation index*	*WOA*	*COA*	*ICOA*
F_1	*mean*	$6.79E+05$	$1.90E+03$	$4.38E+02$
	std	$5.64E+04$	$2.63E+02$	$1.05E+01$
	min	$1.15E+01$	$1.11E-11$	$0.00E+00$
F_2	*mean*	$6.11E+03$	$6.88E+00$	$1.49E+00$
	std	$5.11E+02$	$1.50E+00$	$4.66E-02$
	min	$3.12E+01$	$7.17E-22$	$0.00E+00$
F_3	*mean*	$1.54E+02$	$1.34E+02$	$3.02E+01$
	std	$4.70E+02$	$1.97E+01$	$7.25E-01$
	min	$0.00E+00$	$4.50E-03$	$0.00E+00$
F_4	*mean*	$4.35E+02$	$5.87E+00$	$5.63E-01$
	std	$2.48E+03$	$3.44E+00$	$2.24E-02$
	min	$6.88E+01$	$1.83E-04$	$8.88E-16$

5.3. **Comparison of two sliding mode prediction algorithms.** In this section, we respectively apply the method of [25], and traditional COA to the model of this paper, and all other conditions remain the same. Compared with [25] and COA, when the state delay and input delay are respectively set as $\tau_1 = 5$ and $\tau_2 = 3$, the experiment results shown in Figure 1 to Figure 5 indicate that the algorithm designed in this paper is better in control accuracy, convergence speed, and robust stability.

It can be seen in Figure 1, the position trajectories of X-axis, the performances of the algorithm in this paper are superior to the method in [25] and traditional COA, and at about $k = 175$, the quad-rotor helicopter tends to be smooth and steady in the event of actuator failures.

The results of the comparison curves in Figure 2 and Figure 3 show that when actuator faults occur, the algorithm designed in this paper is better than the algorithm in [25] and COA, not only, for weakening chattering, but also, for accelerating the process of stabilization. Especially, from Figure 3, the trajectories of control law, we can conclude that the ICOA designed in this paper effectively speeds up convergence speed and reduces local extremes in the early stage of control.

Besides, it can be seen in Figure 4 that, the application of the improved double-power function with an improved fault compensation term significantly weakens system chattering and improves control accuracy.

In dealing with time delays, we need to further illustrate the effectiveness of the method in this paper by separately discussing two cases. In the first experiment, we set state delay

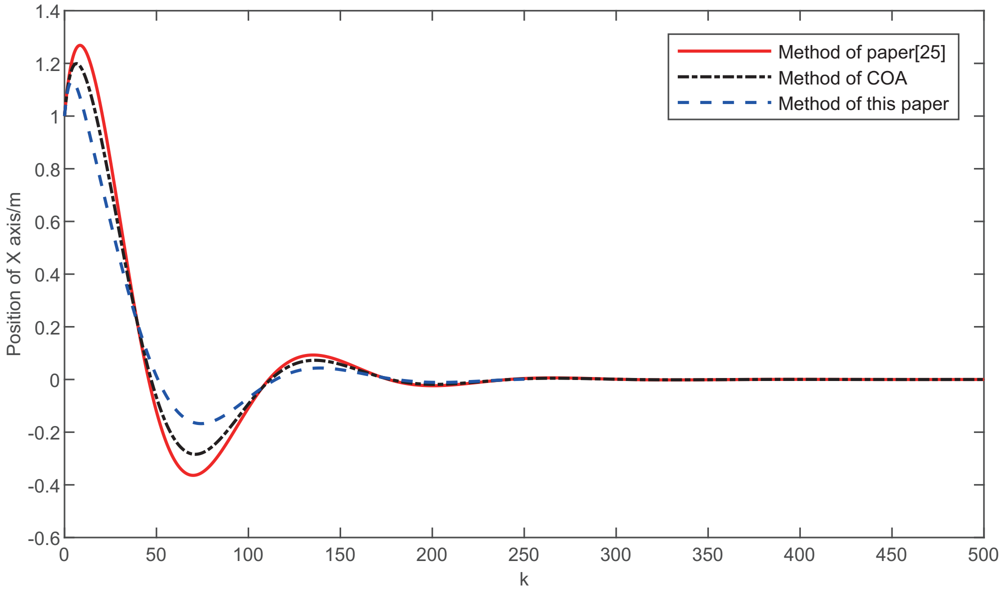

Figure 1. The position trajectories of X-axis

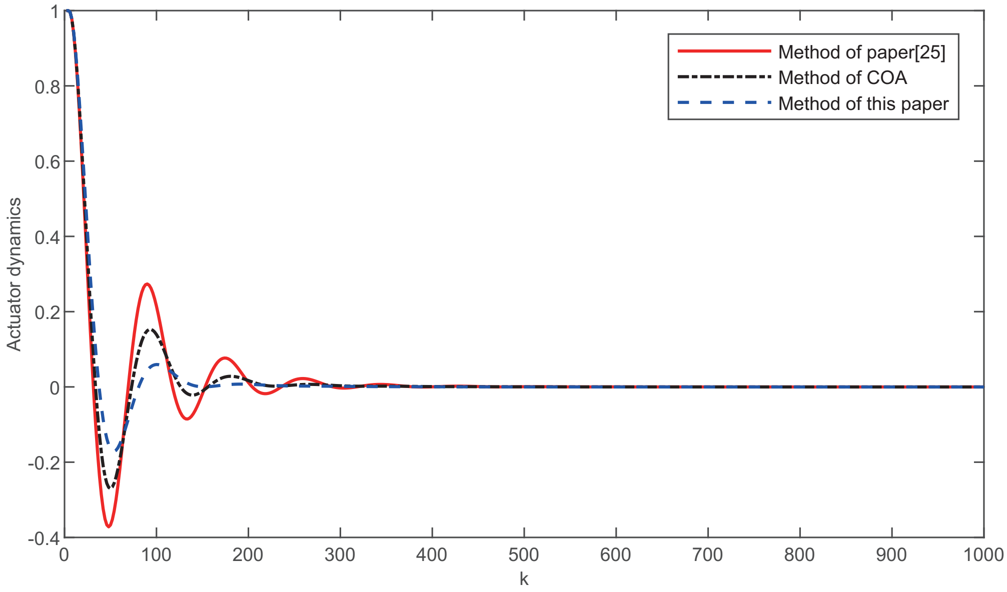

Figure 2. The actuator dynamics trajectories of X-axis

and input delay as $\tau_1 = 0.8$ and $\tau_2 = 0.8$, that means both delays are taken quite small. In Figure 5, we can get that when both delays are small, the method in [25] and the traditional COA can also have nice performance.

From the comparison curves above between the method designed in this paper, the method in [25], and the traditional COA, we can prove the method's superiority and feasibility designed in this paper.

6. **Conclusion.** A sliding mode prediction fault-tolerant control method based on ICOA is designed in this paper. To eliminate the approaching mode and ensure global robustness, we design a quasi-integral sliding mode switching function as a prediction model.

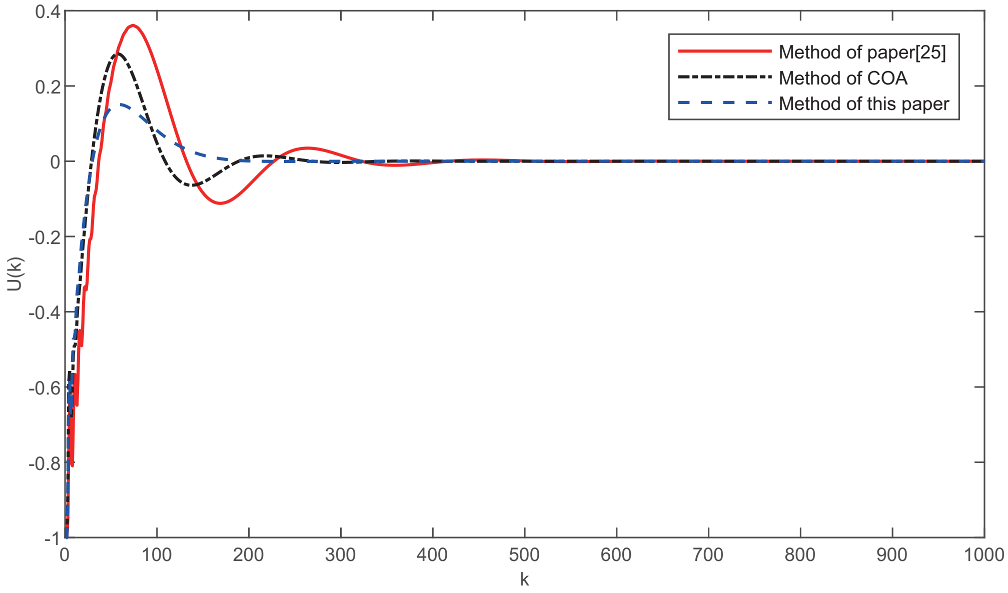

Figure 3. The trajectories of control law

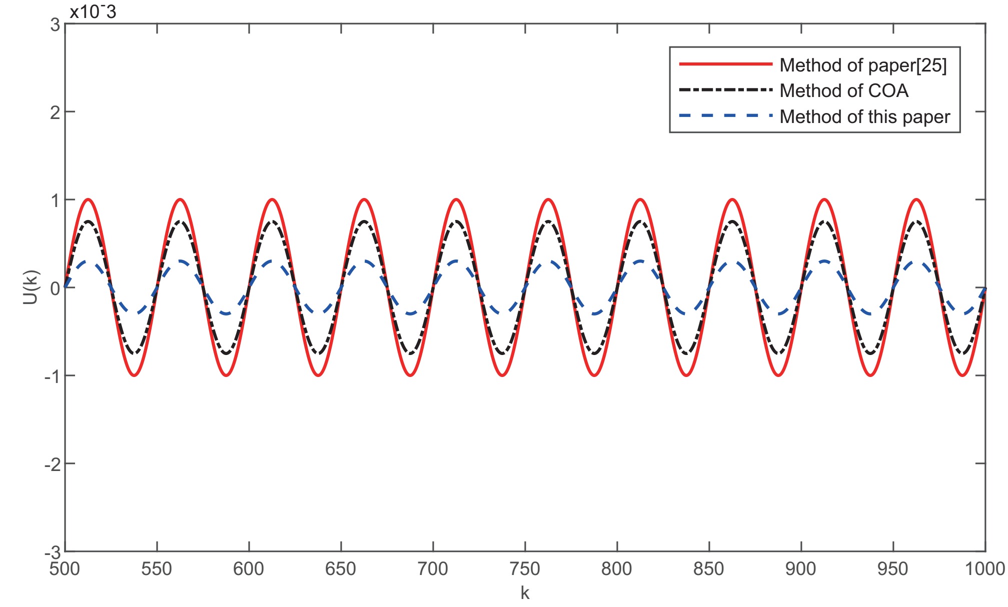

Figure 4. Enlarged view of control law trajectories

Aiming at actuator faults and multiple time delays, the improved double-power function reference trajectory is designed to reduce the time lags's impact on system, and an improved fault compensation term is considered to improve control accuracy. Moreover, the ICOA designed in this paper effectively accelerates convergence speed. Ultimately, the feasibility of the proposed control algorithm is successfully proved by a simulation study on the quad-rotor.

Acknowledgment. The authors gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation. This work is supported by the National Key Laboratory of Science and Technology on Helicopter Transmission

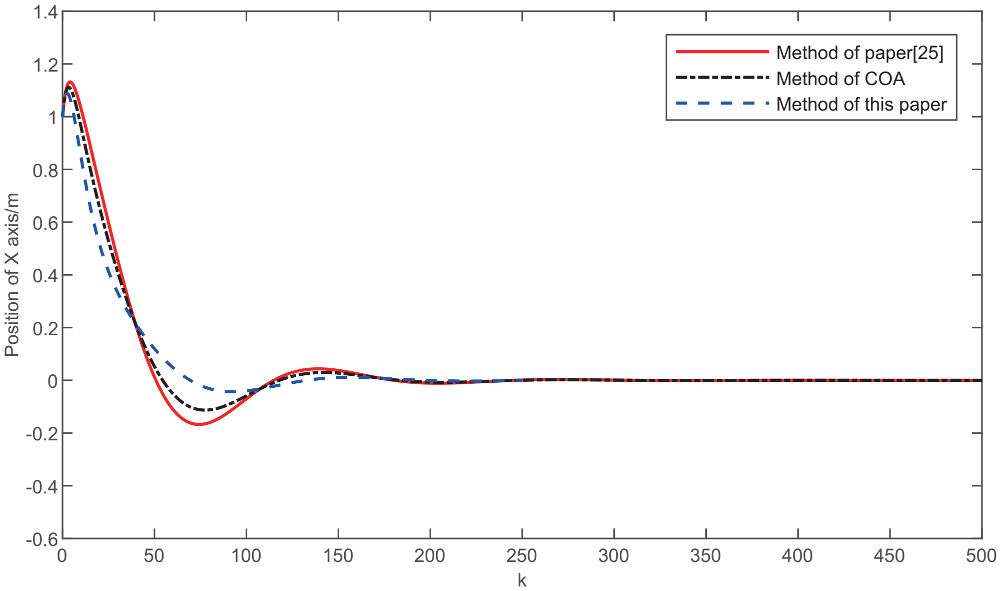

Figure 5. The position trajectories of X-axis (2)

Nanjing University of Aeronautics and Astronautics (HTL-O-19G11), the National Natural Science Foundation of China (61922042) and the Fundamental Research Funds for the Central Universities (Nos. NJ2019014, NJ2020004).

REFERENCES

[1] S. H. Jeong and S. Jung, A quad-rotor system for driving and flying missions by tilting mechanism of rotors: From design to control, *Mechatronics*, vol.24, no.8, pp.20-33, 2014.

[2] X. Dai, Y. Mao, T. Huang, N. Qin, D. Huang and Y. Li, Automatic obstacle avoidance of quadrotor UAV via CNN-based learning, *Neurocomputing*, vol.402, pp.346-358, 2020.

[3] D. Wang, W. Li, X. Liu, N. Li and C. Zhang, UAV environmental perception and autonomous obstacle avoidance: A deep learning and depth camera combined solution, *Computers and Electronics in Agriculture*, vol.175, 2020.

[4] X. Ai and J. Yu, Flatness-based finite-time leader-follower formation control of multiple quadrotors with external disturbances, *Aerospace Science and Technology*, vol.92, pp.20-33, 2019.

[5] Z. Cai, J. Lou, J. Zhao, K. Wu, N. Liu and Y. Wang, Quadrotor trajectory tracking and obstacle avoidance by chaotic grey wolf optimization-based active disturbance rejection control, *Mechanical Systems and Signal Processing*, vol.128, pp.636-654, 2019.

[6] H. Liu, Y. Tian, F. L. Lewis, Y. Wan and K. P. Valavanis, Robust formation tracking control for multiple quadrotors under aggressive maneuvers, *Automatica*, vol.105, pp.179-185, 2019.

[7] H. Liu, M. Lin and L. Deng, UAV route planning for aerial photography under interval uncertainties, *Optik*, vol.127, no.20, pp.9695-9700, 2016.

[8] Z. H. Mao and B. Jiang, Fault identification and fault-tolerant control for a class of networked control systems, *International Journal of Innovative Computing, Information and Control*, vol.3, no.5, pp.1121-1130, 2007.

[9] M. Yahara, K. Fujimoto and D. Nishiguchi, A reset type 2-mode digital FLL with anti-pseudo-lock function and phase control, *ICIC Express Letters, Part B: Applications*, vol.10, no.10, pp.903-910, 2019.

[10] J. Zhang, N. Zhang, G. Shen et al., Analysis and design of chattering-free discrete-time sliding mode control, *International Journal of Robust and Nonlinear Control*, vol.29, no.18, pp.6572-6581, 2019.

[11] B. Akmee, J. M. Carson and D. S. Bayard, A robust model predictive control algorithm for incrementally conic uncertain/nonlinear systems, *International Journal of Robust and Nonlinear Control*, vol.21, no.5, pp.563-590, 2011.

[12] M. Hernandez-Gonzalez, M. V. Basin and E. A. Hernandez-Vargas, Discrete-time high-order neural network identifier trained with high-order sliding mode observer and unscented Kalman filter, *Neurocomputing*, 2019.

[13] C. Li, S. Tong and Y. Wang, Fuzzy adaptive fault tolerant sliding mode control for SISO nonlinear systems, *International Journal of Innovative Computing, Information and Control*, vol.4, no.12, pp.3375-3383, 2008.

[14] Q. Xu, Digital sliding mode prediction control of piezoelectric micro/nanopositioning system, *IEEE Trans. Control Systems Technology*, vol.23, no.1, pp.297-304, 2015.

[15] X. Gao and Y. Weng, Chattering-free model free adaptive sliding mode control for gas collection process with data dropout, *Journal of Process Control*, vol.93, pp.1-13, 2020.

[16] M. H. Amoozgar, A. Chamseddine and Y. Zhang, Experimental test of a two-stage Kalman filter for actuator fault detection and diagnosis of an unmanned quadrotor helicopter, *Journal of Intelligent and Robotic Systems*, vol.70, pp.107-117, 2013.

[17] W. Gong, B. Li, Y. Yang, H. Ban and B. Xiao, Fixed-time integral-type sliding mode control for the quadrotor UAV attitude stabilization under actuator failures, *Aerospace Science and Technology*, vol.95, 2019.

[18] P. Yang, R. Guo and X. Pan, Study on the sliding mode fault tolerant predictive control based on multi agent particle swarm optimization, *International Journal of Control, Automation and Systems*, vol.15, no.5, pp.2034-2042, 2017.

[19] B. Wang, Y. Shen and Y. Zhang, Active fault-tolerant control for a quadrotor helicopter against actuator faults and model uncertainties, *Aerospace Science and Technology*, vol.99, 2020.

[20] Q. Shu, P. Yang, Y. Wang et al., Fault tolerant predictive control based on discrete-time sliding mode observer for quadrotor UAV, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol.22, no.4, pp.498-505, 2018.

[21] S. Xie, C. Chen, L. Zhou et al., Sliding mode optimum predictive control for aero-engine based on time-delay compensator, *Control Engineering of China*, vol.19, no.5, pp.787-789, 2012 (in Chinese).

[22] S. Mobayen, F. Bayat, H. Omidvar et al., Robust global controller design for discrete-time descriptor systems with multiple time varying delays, *International Journal of Robust and Nonlinear Control*, vol.30, no.7, pp.2809-2831, 2020.

[23] Q. Fu, J. Cai, S. Zhong et al., Input-to-state stability of discrete-time memristive neural networks with two delay components, *Neurocomputing*, vol.329, pp.1-11, 2019.

[24] C. Tan, L. Yang, F. Zhang et al., Stabilization of discrete time stochastic system with input delay and control dependent noise, *Systems and Control Letters*, vol.123, pp.62-68, 2019.

[25] Z. Liu, P. Yang, D. Li and M. Xu, Sliding mode prediction fault-tolerant control method based on whale optimization algorithm, *International Journal of Innovative Computing, Information and Control*, vol.15, no.6, pp.2119-2133, 2019.

[26] K. Abidi, J. X. Xu and X. Yu, On the discrete-time integral sliding-mode control, *IEEE Trans. Automatic Control*, vol.52, no.4, pp.709-715, 2007.

[27] S. Qu, X. Xia and J. Zhang, Dynamics of discrete-time sliding-mode-control uncertain systems with a disturbance compensator, *IEEE Trans. Industrial Electronics*, vol.61, no.7, pp.3502-3510, 2014.