

Article

Event-Triggered Fault-Tolerant ADRC for Variable-Load Quadrotor with Prescribed Performance

Zhichen Li *, Qiaoran Wang  and Huaicheng Yan

Key Laboratory of Smart Manufacturing in Energy Chemical Process of the Ministry of Education, East China University of Science and Technology, Shanghai 200237, China; wang_qr214@163.com (Q.W.); hcyan@ecust.edu.cn (H.Y.)

* Correspondence: zcli@ecust.edu.cn

Abstract

This study proposes an event-triggered fault-tolerant active disturbance rejection control (ADRC) method for variable-load quadrotors with prescribed performance. The quadrotor, as a nonlinear and underactuated system, faces challenges such as payload variations, actuator faults, and external disturbances, which degrade trajectory tracking accuracy and stability. The proposed approach integrates a cascaded ADRC framework, decoupling the system into position and velocity subsystems, each equipped with extended state observers (ESOs) for real-time disturbance estimation and compensation. To enhance robustness, prescribed performance functions dynamically constrain tracking errors within predefined bounds, while event-triggered mechanisms reduce computational load through condition-based updates of control signals. Additionally, a particle swarm optimization (PSO) algorithm is employed for online parameter tuning, improving adaptability. Theoretical analysis confirms the system stability, and simulation results demonstrate the controller effectiveness in handling actuator faults and variable payloads, ensuring accurate trajectory tracking and reduced resource consumption. The method offers a promising solution for robust and efficient quadrotor control in complex environments.

Keywords: active disturbance rejection control; particle swarm optimization algorithm; variable-load quadrotor; trajectory tracking control; event triggered; fault tolerant



Academic Editor: Rui Araújo

Received: 29 May 2025

Revised: 20 June 2025

Accepted: 20 June 2025

Published: 22 June 2025

Citation: Li, Z.; Wang, Q.; Yan, H. Event-Triggered Fault-Tolerant ADRC for Variable-Load Quadrotor with Prescribed Performance. *Appl. Sci.* **2025**, *15*, 7021. <https://doi.org/10.3390/app15137021>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the rapid development of unmanned aerial vehicle (UAV) technology, quadrotors show broad application prospects in military and civil fields. In the military field [1,2], quadrotors are widely used in reconnaissance and surveillance, target localization, electronic countermeasures, and other tasks; in the civil field [3–5], they play an increasingly important role in aerial photography, agriculture, logistics and delivery, emergency rescue, and other aspects.

However, the quadrotor is inherently a strongly coupled nonlinear system with under-driven characteristics. Since quadrotors usually work in complex outdoor environments, they are inevitably affected by a variety of disturbing factors, including signal noise [6], bad weather [7], and so on.

During mission execution, quadrotors often carry varying payloads to meet diverse requirements. As shown in Figure 1, firefighting missions require water tanks, agricultural spraying demands pesticide containers, and logistics delivery involves transporting parcels. The payload variations significantly alter the mass parameters while inducing dynamic

changes in the center of gravity and moment of inertia. These alterations affect dynamic characteristics and compromise trajectory tracking accuracy and attitude stability. Therefore, it is of great practical significance to study the stability control method of quadrotor UAVs under variable load conditions.



Figure 1. Quadrotors with different payloads.

Recent years have witnessed significant advances in quadrotor control research, primarily encompassing conventional control methods [8–10], modern control approaches [11,12], and intelligent control techniques [13,14]. variable-load quadrotors face control difficulties due to changing mass, shifting center of gravity, and external disturbances. Current studies primarily address these challenges through parameter adaptation and disturbance observation methods.

Regarding parameter adaptive control, the scholars of [15] proposed a hybrid fuzzy sliding mode control method to ensure quadrotor stability, though its fuzzy rule processing requires considerable computation time. The study of [16] proposed a mass estimation method using least squares and sliding mode observers. However, this solution only works for situations involving gradual mass reduction during payload release, which limits its practical use. In disturbance observation and compensation, the study of [17] developed an extended state observer (ESO)-based control scheme for quadrotors carrying cable-suspended payloads, achieving effective swing reduction and improved disturbance resistance during lifting maneuvers.

Quadrotors possess constrained onboard computing capabilities as flight platforms. Their control algorithms must therefore maintain strict real-time performance while ensuring computational efficiency. This creates a critical challenge in quadrotor control: how to effectively reduce computational and communication loads while maintaining control performance. event-triggered control (ETC), as an innovative control method, has gained wide attention in the UAV field [18–20]. event-triggered control offers distinct advantages through its unique triggering mechanism that enables seamless integration with various control methods, maintaining system stability while reducing computational and communication resource demands. The scholars of [21] developed a switched-threshold event-triggered mechanism integrated with an ESO, enabling intermittent velocity/disturbance observation and precise estimation while establishing a disturbance-rejection output feedback control law for trajectory tracking with guaranteed Zeno-free operation.

Quadrotors may encounter various actuator failures during mission execution. Fault tolerant control [22–25] means that the closed-loop system remains stable and meets certain performance specifications in the event of the failure of a sensor, actuator or component of the control system. The research of [26] proposes a real-time actuator fault detection and isolation (FDI) method for quadrotor UAVs using a linear parameter-varying model and directional residuals to balance fault sensitivity and disturbance robustness. The scholars of [27] presented a fault-tolerant tracking control method for quadrotors, combining recursive least squares-based motor parameter identification, an augmented state observer, and nonlinear model predictive control (NMPC) to handle actuator faults and disturbances. The study of [28] developed an output-feedback dynamic surface sliding mode passive fault-

tolerant control method for quadrotors under actuator controllable failures, significantly enhancing system robustness against both faults and uncertainties.

Active disturbance rejection control (ADRC) has attracted significant attention due to its unique disturbance estimation and compensation mechanism. This innovative control strategy employs an ESO to achieve the real-time estimation and compensation of both internal and external disturbances [29]. Compared with conventional control methods, ADRC demonstrates distinct advantages including model independence, strong robustness, and enhanced disturbance rejection capabilities.

Significant theoretical advancements were achieved in ADRC. Observer design evolved from the original nonlinear ESO to linear ESO [30,31], adaptive ESO [32,33], fuzzy ESO [34–36], and neural network-based ESO [37]. The innovations of control algorithm integrated ADRC with sliding mode control [38], model predictive control [39], and adaptive control [40], creating hybrid strategies with improved dynamic performance and robustness. Meanwhile, intelligent parameter tuning techniques employing particle swarm optimization (PSO) [41], grey wolf optimization [42], and reinforcement learning [43] enhanced the practicality of ADRC in engineering applications.

This study proposes an event-triggered ADRC for variable-load quadrotors with actuator failures, introducing PSO for online parameter tuning. This study makes the following key contributions:

1. A cascaded ADRC approach is developed by decoupling the system into position and velocity subsystems, each with dedicated ESOs and controllers. Combined with the event triggered mechanism, it effectively reduces the consumption of computing resources.
2. To address actuator failures, a fault-tolerant control method integrating prescribed performance functions with ADRC is proposed, which dynamically constrains tracking errors within predefined bounds through an error transformation mechanism.
3. The controller introduces PSO for online parameter adaptation, enhancing both system adaptability and robustness.

2. Modeling of Variable-Load Quadrotor

To accurately describe the motion state of the quadrotor UAV, two coordinate systems were established as shown in Figure 2: the body coordinate frame $B(o_b, x_b, y_b, z_b)$ and the inertial coordinate frame $E(o_e, x_e, y_e, z_e)$.

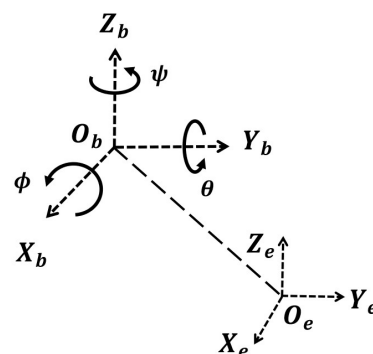


Figure 2. Body and inertial coordinate frame for quadrotor.

The system possesses six degrees of freedom containing both position coordinates (x, y, z) and attitude angles (ϕ, θ, ψ) . These Euler angles represent roll about x_e , pitch about

y_e , and yaw about z_e axes, with counterclockwise rotation defined as positive. The rotation matrix R_{B-E} can be expressed as

$$R_{B-E} = (R_\phi R_\theta R_\psi)^T, \quad (1)$$

where

$$R_\phi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}, R_\theta = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}, R_\psi = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

According to the cross structure of the frame, the quadrotor UAV could be categorized into the “+” type and “×” type. The model in this study was built based on the “×” type structure, which is shown in Figure 3.

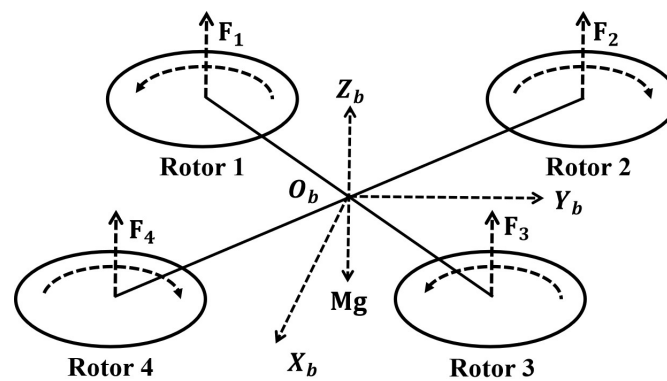


Figure 3. Structure diagram of the quadrotor.

The following assumptions were made for modeling the dynamics:

Assumption 1. The airframe of quadrotor aircraft is typically constructed from lightweight yet rigid materials such as carbon fiber or aluminum alloy. Under normal operational loads, the elastic deformation of these structures can be considered negligible. The quadrotor is rigidly attached and exhibits no elastic deformation.

Assumption 2. Most quadrotors are symmetrically designed. The load-free quadrotor has a uniformly distributed mass, with its center of gravity aligned with the of the body frame.

Assumption 3. The variable payload is rigidly connected to the quadrotor.

Remark 1. The structural rigidity assumption may lose validity under extreme operational conditions, particularly when subjected to high-frequency vibrations, impact loads, or substantial deformations. Assumption 3 considered a rigid connection between the payload and the quadrotor airframe, with non-rigid coupling scenarios explicitly excluded. This modeling approach presents certain limitations.

Based on the physical structure of the quadrotor, it can be observed that the aircraft is primarily influenced by two forces: the lift forces generated by the four rotors and the gravity. The expression for gravitational force in the inertial frame is

$$M_e = \begin{bmatrix} 0 & 0 & (M + m)g \end{bmatrix}^T, \quad (2)$$

where M is the mass of the quadrotor body, m is the mass of the load, and g is the gravitational acceleration.

In the body coordinate system, the lift of the rotor can be expressed as

$$F_b = \begin{bmatrix} 0 & 0 & F_l \end{bmatrix}^T = \begin{bmatrix} 0 & 0 & \sum_{i=1}^4 F_i \end{bmatrix}^T, \quad (3)$$

where F_l denotes the total lift of the rotor and $F_i (i = 1, \dots, 4)$ denotes the lift provided by a single rotor.

Combining Equations (1) and (3), the expression for lift in the inertial coordinate system can be obtained:

$$F_e = \begin{bmatrix} F_x & F_y & F_z \end{bmatrix}^T = R_{B-E} F_b = F_l \begin{bmatrix} \cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi \\ \sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi \\ \cos \theta \cos \phi \end{bmatrix}. \quad (4)$$

During flight, the air resistance experienced by a quadrotor is closely related to its flight velocity, which can be expressed as

$$F_a = \begin{bmatrix} k_a \dot{x} & k_a \dot{y} & k_a \dot{z} \end{bmatrix}^T, \quad (5)$$

where k_a is the air resistance coefficient. To simplify the analysis, the air resistance coefficient was set to be the same in all directions.

By treating the quadrotor UAV and its payload as a single rigid body, according to Newton's law, the displacement dynamics of the variable-load quadrotor in the inertial coordinate system E can be derived as

$$\begin{cases} \ddot{x} = (F_x - k_a \dot{x}) / (M + m) \\ \ddot{y} = (F_y - k_a \dot{y}) / (M + m) \\ \ddot{z} = (F_z - k_a \dot{z}) / (M + m) - g \end{cases}. \quad (6)$$

For the rotational dynamics of the quadrotor, the moment expression for the cross-configuration can be formulated as

$$\tau = \begin{bmatrix} \tau_x & \tau_y & \tau_z \end{bmatrix}^T = \begin{bmatrix} l(-F_1 + F_2 + F_3 - F_4) / \sqrt{2} \\ l(F_1 + F_2 - F_3 - F_4) / \sqrt{2} \\ k_c(F_1 - F_2 + F_3 - F_4) \end{bmatrix}, \quad (7)$$

where l is the length of the rotor arm and k_c is the moment coefficient.

We set $\begin{bmatrix} p & q & r \end{bmatrix}^T$ as the rotational velocity in the body coordinate system. The shifting center of mass during flight transforms the quadrotor into an asymmetric structure, resulting in time-varying moment of inertia matrices. For a quadrotor with variable payload, its inertia matrix can be expressed as

$$J_v = \begin{bmatrix} J_{xx} & -J_{xy} & -J_{xz} \\ -J_{yx} & J_{yy} & -J_{yz} \\ -J_{zx} & -J_{zy} & J_{zz} \end{bmatrix}, \quad (8)$$

where

$$\begin{cases} J_{xx} = J_x + m(l_y^2 + l_z^2) \\ J_{yy} = J_y + m(l_x^2 + l_z^2) \\ J_{zz} = J_z + m(l_x^2 + l_y^2) \\ J_{xy} = J_{yx} = ml_x l_y \\ J_{xz} = J_{zx} = ml_x l_z \\ J_{yz} = J_{zy} = ml_y l_z \end{cases},$$

l_x , l_y , and l_z denote the offset distance between the center of gravity and the center of the body.

When the quadrotor maintains stable flight with minimal attitude variations, the Euler angle rates in the inertial frame can be approximated by the body-frame angular velocities: $\begin{bmatrix} \dot{\phi} & \dot{\theta} & \dot{\psi} \end{bmatrix}^T \approx \begin{bmatrix} p & q & r \end{bmatrix}^T$. This linear approximation permits derivation of the quadrotor rotational dynamics through the Euler equations. The rotational dynamics of the quadrotor can be derived from Euler equations of motion as

$$\begin{cases} \ddot{\phi} = \tau_x / J_{xx} + [J_{yz}(\dot{\theta}^2 - \dot{\psi}^2) + (J_{yy} - J_{zz})\dot{\theta}\dot{\psi} + J_{xz}(\dot{\phi}\dot{\theta} + \ddot{\psi}) + J_{xy}(\ddot{\theta} - \dot{\phi}\dot{\psi})] / J_{xx} \\ \ddot{\theta} = \tau_y / J_{yy} + [J_{xz}(\dot{\psi}^2 - \dot{\phi}^2) + (J_{zz} - J_{xx})\dot{\phi}\dot{\psi} + J_{xy}(\dot{\theta}\dot{\psi} + \ddot{\phi}) + J_{yz}(\ddot{\psi} - \dot{\phi}\dot{\theta})] / J_{yy} \\ \ddot{\psi} = \tau_z / J_{zz} + [J_{xy}(\dot{\phi}^2 - \dot{\theta}^2) + (J_{xx} - J_{yy})\dot{\phi}\dot{\theta} + J_{yz}(\dot{\phi}\dot{\psi} + \ddot{\theta}) + J_{xz}(\ddot{\phi} - \dot{\theta}\dot{\psi})] / J_{zz} \end{cases} \quad (9)$$

The displacement Equation (6) and rotational Equation (9) collectively constitute the dynamic model of the variable-load quadrotor.

By introducing virtual control variables U_1, U_2, U_3, U_4 into the control system, where $\begin{bmatrix} U_1 & U_2 & U_3 & U_4 \end{bmatrix}^T = \begin{bmatrix} F_l & \tau_x & \tau_y & \tau_z \end{bmatrix}^T$, and then combining Equations (3) and (7), the control inputs to the quadrotor can be obtained as

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} F_l \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -\frac{l}{\sqrt{2}} & \frac{l}{\sqrt{2}} & \frac{l}{\sqrt{2}} & -\frac{l}{\sqrt{2}} \\ \frac{l}{\sqrt{2}} & \frac{l}{\sqrt{2}} & \frac{l}{\sqrt{2}} & \frac{l}{\sqrt{2}} \\ k_c & -k_c & k_c & -k_c \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix}. \quad (10)$$

The control model of the variable-load quadrotor can be given as

$$\begin{cases} \ddot{x} = [U_1(\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi) - k_a \dot{x}] / (M + m) \\ \ddot{y} = [U_1(\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi) - k_a \dot{y}] / (M + m) \\ \ddot{z} = [U_1(\cos \theta \cos \phi) - k_a \dot{z}] / (M + m) - g \\ \ddot{\phi} = [U_2 + ml_y l_z(\dot{\theta}^2 - \dot{\psi}^2) + (J_y - J_z + ml_z^2 - ml_y^2)\dot{\theta}\dot{\psi} + ml_x l_z(\dot{\phi}\dot{\theta} + \ddot{\psi}) + ml_x l_y(\ddot{\theta} - \dot{\phi}\dot{\psi})] / [J_x + m(l_y^2 + l_z^2)] \\ \ddot{\theta} = [U_3 + ml_x l_z(\dot{\psi}^2 - \dot{\phi}^2) + (J_z - J_x + ml_x^2 - ml_z^2)\dot{\phi}\dot{\psi} + ml_x l_y(\dot{\theta}\dot{\psi} + \ddot{\phi}) + ml_y l_z(\ddot{\psi} - \dot{\phi}\dot{\theta})] / [J_y + m(l_x^2 + l_z^2)] \\ \ddot{\psi} = [U_4 + ml_x l_y(\dot{\phi}^2 - \dot{\theta}^2) + (J_x - J_y + ml_y^2 - ml_x^2)\dot{\phi}\dot{\theta} + ml_y l_z(\dot{\phi}\dot{\psi} + \ddot{\theta}) + ml_x l_z(\ddot{\phi} - \dot{\theta}\dot{\psi})] / [J_z + m(l_x^2 + l_y^2)] \end{cases} \quad (11)$$

The actuator system, comprising four electronic speed controllers (ESCs) with corresponding brushless motor-propeller assemblies, serves as the critical actuation core in quadrotor flight control. During operational missions, various actuator failures may occur due to mechanical wear, electronic component malfunctions, or external disturbances, which directly alter the dynamic characteristics of the quadrotor.

In quadrotor systems, actuator faults typically manifest as the following:

- **Loss of Effectiveness:** Manifested as a decrease in motor thrust output efficiency; potential causes include insufficient power supply voltage, reduced ESC drive capability, or propeller damage.

- Bias Fault: Characterized by a constant offset in motor output; potential causes include sensor zero drift or ESC calibration errors.

When the actuator has the above faults, the actual lift it provides can be written as

$$F_i^F = a_i^F F_i + b_i^F, \quad (12)$$

where F_i^F denotes the actual lift force generated by the actuator. The efficiency factor $a_i^F \in [0, 1]$ represents the effectiveness of the actuator, while b_i^F indicates the bias factor. The system operates under the following conditions: when $a_i^F = 1$ and $b_i^F = 0$, the quadrotor is in normal operation; when $a_i^F \in [0, 1)$ and $b_i^F = 0$, the quadrotor has loss of effectiveness in the actuator; and when $a_i^F = 1$ with $b_i^F \neq 0$, a bias fault is present in the system.

According to Equations (10) and (12), the actual control inputs of the variable-load quadrotor in the case of actuator failure can be obtained as

$$\begin{bmatrix} F_l^F \\ \tau_x^F \\ \tau_y^F \\ \tau_z^F \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -\frac{\sqrt{2}}{2}l & \frac{\sqrt{2}}{2}l & \frac{\sqrt{2}}{2}l & -\frac{\sqrt{2}}{2}l \\ \frac{\sqrt{2}}{2}l & \frac{\sqrt{2}}{2}l & -\frac{\sqrt{2}}{2}l & -\frac{\sqrt{2}}{2}l \\ k_c & -k_c & k_c & -k_c \end{bmatrix} \begin{bmatrix} F_1^F \\ F_2^F \\ F_3^F \\ F_4^F \end{bmatrix}. \quad (13)$$

Combining Equations (6)–(13), the dynamic model of the variable-load quadrotor with actuator faults can be obtained as

$$\begin{cases} \ddot{x} = [F_l^F (\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi) - k_a \dot{x}] / (M + m) \\ \ddot{y} = [F_l^F (\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi) - k_a \dot{y}] / (M + m) \\ \ddot{z} = [F_l^F (\cos \theta \cos \phi) - k_a \dot{z}] / (M + m) - g \\ \ddot{\phi} = \tau_x^F / J_{xx} + [J_{yz}(\dot{\theta}^2 - \dot{\psi}^2) + (J_{yy} - J_{zz})\dot{\theta}\dot{\psi} + J_{xz}(\dot{\phi}\dot{\theta} + \ddot{\psi}) + J_{xy}(\ddot{\theta} - \dot{\phi}\dot{\psi})] / J_{xx} \\ \ddot{\theta} = \tau_y^F / J_{yy} + [J_{xz}(\dot{\psi}^2 - \dot{\phi}^2) + (J_{zz} - J_{xx})\dot{\phi}\dot{\psi} + J_{xy}(\dot{\theta}\dot{\psi} + \ddot{\phi}) + J_{yz}(\ddot{\psi} - \dot{\phi}\dot{\theta})] / J_{yy} \\ \ddot{\psi} = \tau_z^F / J_{zz} + [J_{xy}(\dot{\phi}^2 - \dot{\theta}^2) + (J_{xx} - J_{yy})\dot{\phi}\dot{\theta} + J_{yz}(\dot{\phi}\dot{\psi} + \ddot{\theta}) + J_{xz}(\ddot{\phi} - \dot{\theta}\dot{\psi})] / J_{zz} \end{cases}. \quad (14)$$

The inherent underactuation and strong dynamic coupling in quadrotor systems mean that any actuator fault propagates disturbances across all 6-degree-of-freedom motion states. When a fault occurs, the actual actuator lift force F_i^F is obtained through Equation (12). Using Equation (13), the total lift and individual moments under actuator fault conditions can be computed, which drive the flight through the dynamics model (14).

3. Design of Event-Triggered Cascade Active Disturbance Rejection Control with Prescribed Performance

The quadrotor represents an underactuated system with four inputs governing six degrees of freedom. This configuration inherently prevents complete decoupling within the system. Through a cascaded outer-inner loop structure, the controller achieves implicit decoupling between positional and attitudinal kinematic relationships. Displacement control along the x - and y -axes must be achieved through attitude angle regulation.

Based on this principle, we propose a dual-loop cascade active disturbance rejection control (CADRC) scheme with prescribed performance control (PPC). An event-triggered mechanism is introduced to reduce computational redundancy, while PSO is employed for online controller parameter self-tuning.

The control system structure is shown in Figure 4. The outer-loop position controller generates the desired pitch angle θ_d , roll angle ϕ_d , and U_1 , which are then passed to the inner-loop attitude controller for conversion into U_2 , U_3 , and U_4 .

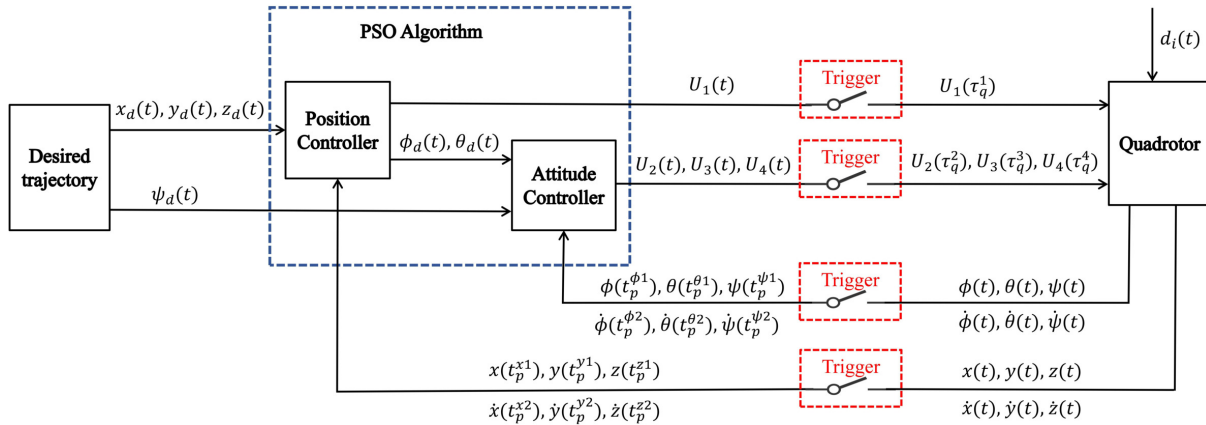


Figure 4. A dual-loop control framework based on event-triggered mechanism for the variable-load quadrotor.

3.1. Prescribed Performance Function and Error Transformation

Actuator faults are considered part of the total disturbance in the ADRC framework. The ESO estimates and compensates for these faults along with other disturbances. To enhance the disturbance rejection capability while mitigating initial overshoot, prescribed performance functions are introduced to further optimize control performance.

Consider the altitude z -channel as an example. A monotonically decreasing performance boundary function $p_z(t)$ is defined to constrain the transient tracking error $e_z = z - z_d$. The boundary function is

$$p_z(t) = (p_{z0} - p_{z\infty})e^{-\mu_z t} + p_{z\infty}, \quad (15)$$

where $p_{z0} > 0$ is the initial value of the prescribed performance function and $p_{z\infty} = \lim_{t \rightarrow \infty} p_z(t) > 0$ is the steady state value of the prescribed performance function, $p_{z0} > p_{z\infty}$. The $p_{z\infty}$ directly determines the ultimate precision of the system. The μ_z is the rate factor for the speed of convergence, which determines how quickly the error boundary decays from its initial to steady-state value. A larger μ_z accelerates convergence but may cause high-frequency oscillations. A smaller μ_z produces smoother responses but extends transition time.

If the system error e_z is satisfied,

$$-\underline{\lambda}_z p_z(t) < e_z < \overline{\lambda}_z p_z(t), \quad (16)$$

then the prescribed performance is satisfied. Here, $\underline{\lambda}_z, \overline{\lambda}_z \in (0, 1]$ are the error boundary parameters to be designed. These parameters control the allowable range of tracking errors. A larger λ_z permits greater instantaneous errors but may reduce steady-state accuracy. A smaller λ_z imposes stricter error constraints but may increase control effort.

The constrained transient error e_z is transformed into an unconstrained error variable ϵ_z through conversion mapping. The error transformation function is designed as follows:

$$e_z(t) = p_z(t) S_z(\epsilon_z), \quad (17)$$

where $S_z(\epsilon_z)$ is a smooth, strictly monotone decreasing function satisfying $S_z(-\infty) = -\underline{\lambda}_z$, $S_z(+\infty) = \overline{\lambda}_z$. The expression for $S_z(\epsilon_z)$ is

$$S_z(\epsilon_z) = \frac{\overline{\lambda}_z e^{\epsilon_z} - \underline{\lambda}_z e^{-\epsilon_z}}{e^{\epsilon_z} + e^{-\epsilon_z}}. \quad (18)$$

We then define $Z = \frac{e_z}{p_z}$, and, by calculating the inverse function of Equation (18), one can obtain an expression for ϵ_z as

$$\epsilon_z = \frac{1}{2} \ln \left(\frac{Z + \lambda_z}{\lambda_z - Z} \right). \quad (19)$$

The derivation of ϵ_z leads to

$$\dot{\epsilon}_z = r_z(\dot{e}_z - Z\dot{p}_z), \quad (20)$$

$$\text{where } r_z = \frac{\bar{\lambda}_z + \underline{\lambda}_z}{2p_z(Z + \underline{\lambda}_z)(\bar{\lambda}_z - Z)}.$$

3.2. Design of an Event-Triggered Cascade Active Disturbance Rejection Controller

The dual-loop control architecture comprises three sub-controllers in each loop: x -, y -, and z -axis controllers in the position loop and roll (ϕ), pitch (θ), and yaw (ψ) controllers in the attitude loop.

Due to the dynamic characteristics of the quadrotor, the sub-controllers are hierarchically organized into position and velocity controllers to construct a cascaded ADRC system. Taking the altitude z channel as an example, the altitude channel dynamics described in model (11) can be rewritten as

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = \zeta z_2 + \eta U_z + f + d_z \end{cases}, \quad (21)$$

where $[z_1 \ z_2]^T = [z \ \dot{z}]^T$, $\zeta = -\frac{k_a}{M+m}$, $\beta = \frac{1}{M+m}$, f represents the other parts in the modeling of the system, and d_z stands for the unmodeled parts of the system and disturbances outside the system.

Defining a new virtual control variable $h_z = z_2$, the dynamics expressed in Equation (21) can be rewritten as

$$\text{Position subsystem } S_{z1}: \dot{\chi}_{z1} = h_z + \omega_{z1}$$

$$\text{Velocity subsystem } S_{z2}: \dot{\chi}_{z2} = U_z + \omega_{z2}$$

where ω_{z1} and ω_{z2} represent the effects of disturbances and system nonlinearities in the two subsystems, respectively. The variable h_z is obtained from the position subsystem S_{z1} . This variable serves as the reference input for the velocity subsystem S_{z2} .

The cascade ADRC controller design consists of two steps:

Step 1: For subsystem S_{z1} , the extended state can be written as

$$\begin{cases} \dot{\alpha}_{z1} = \alpha_{z2} + h_z \\ \dot{\alpha}_{z2} = \omega_{z1} \end{cases}. \quad (22)$$

The ESO is designed as

$$\begin{cases} \dot{\hat{\alpha}}_{z1} = \hat{\alpha}_{z2} + h_z + l_{z1}(\rho_{z1} - \hat{\alpha}_{z1}) \\ \dot{\hat{\alpha}}_{z2} = l_{z2}(\rho_{z1} - \hat{\alpha}_{z1}) \end{cases}, \quad (23)$$

where $\hat{\alpha}_{zi}(i = 1, 2)$ are estimates of the state variable $\alpha_{zi}(i = 1, 2)$; the observer gains $l_{zi}(i = 1, 2)$ are designed to ensure that matrix $A_{z1} = \begin{bmatrix} -l_{z1} & 1 \\ -l_{z2} & 0 \end{bmatrix}$ is Hurwitz. The variable ρ_{z1} represents the sampled value of system state $z_1(t)$ at the event-triggered instant t_p^{z1} ,

which serves as the state value for the current cycle. The event-triggered mechanism for this observer is designed as follows:

$$\begin{cases} \rho_{z1}(t) = z_1(t_p^{z1}), & t \in [t_p^{z1}, t_{p+1}^{z1}) \\ t_{p+1}^{z1} = \inf \left\{ t > t_p^{z1} \mid |z_1(t) - \rho_{z1}(t)| > \gamma_{z1} \right\} \end{cases} \quad (24)$$

The observer trigger threshold γ_{z1} is designed as a fixed value. Based on the desired trajectory z_d , the virtual control variable h_z is designed as follows:

$$h_z = k_{z1}(z_d - \hat{z}_{z1}) - \hat{z}_{z2}, \quad (25)$$

where k_{z1} is the controller parameter to be tuned.

Step 2: For subsystem S_{z2} , the extended state can be written as

$$\begin{cases} \dot{\beta}_{z1} = \beta_{z2} + \sigma_z \\ \dot{\beta}_{z2} = \dot{w}_{z2} \end{cases} \quad (26)$$

The ESO is designed as

$$\begin{cases} \dot{\hat{\beta}}_{z1} = \hat{\beta}_{z2} + \sigma_z + l_{z3}(\rho_{z2} - \hat{\beta}_{z1}) \\ \dot{\hat{\beta}}_{z2} = l_{z4}(\rho_{z2} - \hat{\beta}_{z1}) \end{cases}, \quad (27)$$

where $\hat{\beta}_{zi}(i = 1, 2)$ is an estimate of the state variable $\beta_{zi}(i = 1, 2)$, and the observer gains $l_{zi}(i = 3, 4)$ are required to satisfy the matrix $A_{z2} = \begin{bmatrix} -l_{z3} & 1 \\ -l_{z4} & 0 \end{bmatrix}$, which is Hurwitz. The variable ρ_{z2} represents the sampled value of system state $z_2(t)$ at the event-triggered instant t_p^{z2} . The event-triggered mechanism for this observer is designed as follows:

$$\begin{cases} \rho_{z2}(t) = z_2(t_p^{z2}), & t \in [t_p^{z2}, t_{p+1}^{z2}) \\ t_{p+1}^{z2} = \inf \left\{ t > t_p^{z2} \mid |z_2(t) - \rho_{z2}(t)| > \gamma_{z2} \right\} \end{cases}, \quad (28)$$

where γ_{z2} is designed as a fixed value. Due to the event triggered mechanism, the trajectory tracking error for the z channel is defined as

$$e_z = \rho_{z1} - z_d. \quad (29)$$

According to the virtual control variable h_z and tracking error, the virtual control variable U_z is designed as follows:

$$U_z = k_{z2}(h_z - \hat{\beta}_{z1}) - \hat{\beta}_{z2} + Z\dot{p}_z, \quad (30)$$

where k_{z2} is the controller parameter to be tuned.

Remark 2. The cascaded structure decouples the system into two subsystems, enabling targeted disturbance rejection. The outer-loop ESO (23) estimates position-related disturbances, while the inner-loop ESO (27) handles velocity-level uncertainties. The dual-loop design enables optimal balance between transient response and noise suppression by adjusting bandwidth. It overcomes the inherent limitation of conventional single-ESO systems, which must find a compromise between dynamic performance and disturbance rejection. This approach ultimately improves the overall system robustness.

The control design of the x and y channels in the position controller is similar to that of the z channel. The sampled values of virtual control variables U_x , U_y , and U_z and yaw angle ψ at the event-triggered instant t_p^ψ are utilized to compute the desired roll/pitch angles and control command U_1 .

$$\phi_d = \arctan(\cos \theta(t_q^\theta) \frac{U_x \sin \psi(t_p^\psi) - U_y \cos \psi(t_p^\psi)}{U_z}), \quad (31)$$

$$\theta_d = \arctan(\frac{U_x \cos \psi(t_p^\psi) + U_y \sin \psi(t_p^\psi)}{U_z}), \quad (32)$$

$$U_1 = \frac{U_z}{\cos \phi(t_q^\phi) \cos \theta(t_q^\theta)}. \quad (33)$$

The variables σ_x , σ_y , and σ_z are calculated based on the sampled value of the control variable $U_1(t)$ at the event-triggered instant τ_q^1 . The controller event-triggered mechanism is designed to

$$\begin{cases} \sigma_1(t) = U_1(\tau_q^1), & t \in [\tau_q^1, \tau_{q+1}^1) \\ \sigma_x(t) = \sigma_1(t)(\cos \psi(t_p^\psi) \sin \theta(t_p^\theta) \cos \phi(t_p^\phi) + \sin \psi(t_p^\psi) \sin \phi(t_p^\phi)) \\ \sigma_y(t) = \sigma_1(t)(\sin \psi(t_p^\psi) \sin \theta(t_p^\theta) \cos \phi(t_p^\phi) - \cos \psi(t_p^\psi) \sin \phi(t_p^\phi)) \\ \sigma_z(t) = \sigma_1(t) \cos \phi(t_p^\phi) \cos \theta(t_p^\theta) \\ \tau_{q+1}^1 = \inf \left\{ t > \tau_q^1 \mid |U_1(t) - \sigma_1(t)| > M_1^1 e^{-M_2^1 \hat{e}_t} \right\} \end{cases}, \quad (34)$$

where M_1^1 denotes the threshold extremum factor and M_2^1 represents the error sensitivity factor. The controller trigger threshold is designed as a time-varying value related to the error, and the trajectory estimation error $\hat{e}_t$ is introduced into the design.

$$\hat{e}_t = \sqrt{(\hat{x}_{t1} - x_d)^2 + (\hat{y}_{t1} - y_d)^2 + (\hat{z}_{t1} - z_d)^2}. \quad (35)$$

Remark 3. This study integrates trajectory tracking error into the event-triggered mechanism of the controller, employing a time-varying threshold that dynamically adjusts according to error magnitude. As tracking error grows, the triggering threshold decreases while update frequency rises. Conversely, diminishing error leads to increased threshold values and reduced triggering frequency.

Remark 4. For the quadrotor with strongly coupled dynamics, trajectory tracking error should be computed comprehensively in the 3D space rather than independently. In the error equation (35), the controller cannot obtain the actual state values in real time due to the presence of event-triggered mechanisms. Therefore, the estimated state of the ESO is used for the calculation of the event-trigger mechanisms (24) and (28) and the trajectory tracking error.

This design stems from two key considerations. First, subsequent proofs demonstrate that the estimation error is uniformly ultimately bounded, ensuring the estimated values effectively approximate the true states of the quadrotor. Second, the fundamental objective of event-triggered control lies in minimizing unnecessary computations and communications. Requiring real-time value computation within the controller event-triggered mechanism would contradict this design principle.

The attitude channel sub-controller of the inner loop controller is designed similarly to the position channel. Taking the roll channel as an example, the controller event-triggered mechanism for this channel is

$$\begin{cases} \sigma_2(t) = U_2(\tau_q^2), & t \in [\tau_q^2, \tau_{q+1}^2) \\ \tau_{q+1}^2 = \inf \left\{ t > \tau_q^2 \mid |U_2(t) - \sigma_2(t)| > M_1^2 e^{-M_2^2 |\hat{\alpha}_{\phi 1} - \phi_d|} \right\} \end{cases} \quad (36)$$

Remark 5. This study employs a fixed threshold in the event-triggered mechanism of the observer while adopting a dynamic threshold in the controller. This clearly divided design ensures that the observer rapidly and stably estimates system states and disturbances, with the fixed threshold guaranteeing consistent response speed unaffected by dynamic parameters.

The controller utilizes a dynamic threshold to balance tracking error and computational overhead. This approach ensures adaptability to varying performance demands during different flight phases. The fixed threshold of the observer provides reliable estimated inputs for the controller, while the dynamic threshold of the controller further optimizes overall resource efficiency.

3.3. Cascade Active Disturbance Rejection Controller Based on Particle Swarm Optimization

Recognizing the critical influence of controller parameters on cascaded ADRC system performance, this study develops an online parameter self-tuning method based on PSO, which effectively enhances the adaptability and robustness of the control strategy.

Each cascade sub-controller has six parameters $[l_1, l_2, l_3, l_4, k_1, k_2]$ to be tuned. Thus, the entire inner-loop attitude and outer-loop position control system has a total of 36 adjustable parameters. The bandwidth method helps simplify parameter tuning. This approach utilizes two important parameters: the observer bandwidth ω_o and the controller bandwidth ω_c . The parameters can then be systematically determined through the following bandwidth-based relationships:

$$l_1 = \omega_{o1}^2, \quad l_2 = 2\omega_{o1}, \quad l_3 = \omega_{o2}^2, \quad l_4 = 2\omega_{o2}, \quad k_1 = \omega_c^2, \quad k_2 = 2\omega_c. \quad (37)$$

within the attitude control loop, the three attitude controllers exhibit control objectives of similar magnitude. Likewise, the three position controllers in the displacement control loop maintain consistent scale. This means that identical parameter sets can be employed for each control loop.

The simplification process cuts the total tuning parameters from 36 down to 6. Among these, three parameters handle attitude control while the remaining three manage position control. This reduction maintains effective control performance. The parameters requiring adjustment consist of $\omega_{oa1}, \omega_{oa2}, \omega_{ca}, \omega_{op1}, \omega_{op2}$, and ω_{cp} .

Remark 6. In the controller parameterization process of the quadrotor, the six sub-controllers function as a coupled system. The quadrotor exhibits strong inherent coupling between its displacement and rotational attitude. This interdependence limits how individual sub-controllers can be tuned. Consequently, the six sub-controllers cannot be parameterized completely independently. Consider the case where the yaw angle $\psi = 0$. If the attitude controller performs poorly, it may fail to generate adequate roll angle. This deficiency directly impairs tracking performance along the x-axis. In such cases, optimizing only the x-channel controller parameters fails to effectively improve tracking performance.

In PSO, each particle updates its position and velocity according to the following rules:

$$V_i^{t+1} = wV_i^t + c_1r_1(P_{besti} - X_i^t) + c_2r_2(G_{best} - X_i^t), \quad (38)$$

$$X_i^{t+1} = X_i^t + V_i^{t+1}, \quad (39)$$

where i denotes the i th particle in the swarm, V_i^t denotes the velocity of the i th particle at the t th iteration, X_i^t denotes the position at the t th iteration, P_{besti} denotes the best position in the history of individual particles, G_{best} denotes the global best position, w denotes the inertia weight, c_1 and c_2 denote acceleration constants, and r_1 and r_2 are random numbers between 0 and 1. By iteratively updating the particle positions and velocities, the PSO can achieve effective exploration of the solution space.

The inertia weight w is an important parameter that determines the effect of inertia on the historical velocity of the particle when updating the velocity. A linearly decreasing inertia weighting strategy is introduced to dynamically adjust w , which is updated as follows:

$$w = w_{ini} - \frac{t}{t_{max}}(w_{ini} - w_{fin}), \quad (40)$$

where w_{ini} and w_{fin} are the initial and final values of the inertia weights, respectively; t denotes the current number of iterations; and t_{max} denotes the maximum number of iterations. The algorithm dynamically adjusts parameter w through linear decrement, maintaining balance between global and local exploration throughout the optimization process.

The fitness function plays a crucial role in determining optimization results. It must properly evaluate particle quality to effectively guide the particle swarm's search direction. Overly complex function designs may decrease iteration counts. However, the computational time required for these complex operations could lower overall efficiency. Therefore, the design of the fitness function needs to balance guidance and efficiency.

With the goal of trajectory tracking, the tracking error of the quadrotor on the target trajectory is crucial for evaluating the control performance. For a quadrotor system operating in three dimensions, the trajectory tracking error is defined as

$$e_q = \sqrt{(x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2}. \quad (41)$$

In quadrotor flight control, attitude stability is as important as trajectory tracking accuracy. To quantitatively evaluate the degree of oscillation of the attitude angle, the standard deviation is used as a key indicator. Taking the roll angle ϕ as an example, the standard deviation calculation expression is

$$s_\phi = \sqrt{\frac{1}{n} \sum_{i=1}^n (\phi_i - \bar{\phi})^2}, \quad (42)$$

where n is the number of samples and $\bar{\phi}$ is the sample mean.

The designed fitness function is

$$Y = a \int_0^t \tau e_q(\tau) d\tau + b(s_\phi + s_\theta + s_\psi), \quad (43)$$

where a and b are positive parameters and s_ϕ , s_θ , and s_ψ are the standard deviation of the roll, pitch, and yaw angles, respectively. In the design of this function, time is considered as the weight of the error. The function treats time as an error weighting variable, where increased duration enhances tracking error influence on fitness function. The fitness function output Y inversely relates to controller performance. Lower Y values correspond to better control effectiveness.

The algorithm integrating PSO for online parameter adaptation in cascade ADRC is shown in Algorithm 1.

Algorithm 1: Cascade ADRC Based on PSO

```

1 for each time step  $t$  do
2   if first run or ( $\text{error} > \text{threshold}$  and  $\text{cooldown elapsed}$ ) then
3     Initialize particle swarm
4     for  $i = 1$  to Number of iteration do
5        $w \leftarrow w_{\max} - (w_{\max} - w_{\min}) \frac{i}{n_{\text{iter}}}$ 
6       for each particle  $j$  do
7         Simulate forward  $N$  steps with  $\mathbf{x}_j$ 
8         Compute fitness  $Y_j$ 
9         Update velocity:  $\mathbf{v}_j \leftarrow w\mathbf{v}_j + c_1r_1(\mathbf{p}_j - \mathbf{x}_j) + c_2r_2(\mathbf{g} - \mathbf{x}_j)$ 
10        Update position:  $\mathbf{x}_j \leftarrow \mathbf{x}_j + \mathbf{v}_j$ 
11        Update personal best  $\mathbf{p}_j$  if improved
12      end
13      Update global best  $\mathbf{g}$  if improved
14    end
15    Update controller parameters:  $\theta_{\text{ADRC}} \leftarrow \mathbf{g}$ 
16  end
17  Update event-triggered state  $\mathbf{x}_{\text{trigger}}$ 
18  Compute control:  $\mathbf{u}_k = \text{ADRC}(\theta_{\text{ADRC}}, \mathbf{x}_{\text{trigger}})$ 
19  Apply  $\mathbf{u}_k$ , observe  $\mathbf{y}_{k+1}$ 
20  Compute error  $e_k = \|\mathbf{y}_{k+1} - \mathbf{y}_{\text{ref}}\|$ 
21 end

```

To achieve online self-tuning of the PSO in the controller, an event-triggered mechanism is designed as follows:

$$\begin{cases} \omega(\cdot) = \omega_{oa}, \omega_{ca}, \omega_{op}, \omega_{cp} \\ \omega(\cdot)(t) = \omega(\cdot)(t_p), \quad t \in [t_p, t_{p+1}) \\ t_{p+1} = \inf \left\{ t > t_p + \underline{t} \mid e_q(t) > \gamma_p \right\} \end{cases}, \quad (44)$$

where t_p denotes the trigger instant of parameter adaptation; t_{p+1} denotes the next trigger instant; $\underline{t}$ denotes the minimum trigger interval, equivalent to the cooling time; and γ_p is the fixed threshold of the designed event trigger. The presence of a trigger interval avoids the Zeno phenomenon in the event-triggered mechanism.

Remark 7. The stable flight control of a quadrotor must meet trajectory tracking accuracy and attitude stability requirements. Simulation results demonstrate that omitting attitude angle constraints enables fast trajectory convergence. However, this approach generates substantial high-frequency attitude oscillations. Such oscillations ultimately impair real flight performance.

4. Stability Analysis

4.1. Convergence Analysis of Extended State Observer

We define the estimation error of ESO as $\tilde{\alpha}_i = \alpha_i - \hat{\alpha}_i$, $\tilde{\beta}_i = \beta_i - \hat{\beta}_i$, ($i = 1, 2$). Combining Equations (22), (23), (26), and (27), the dynamic equation of the estimation error system can be obtained as

$$\begin{cases} \dot{\tilde{\alpha}}_1 = \dot{\alpha}_1 - \dot{\hat{\alpha}}_1 = -l_1\tilde{\alpha}_1 + \tilde{\alpha}_2 - l_1(\rho_1 - \alpha_1) \\ \dot{\tilde{\alpha}}_2 = \dot{\alpha}_2 - \dot{\hat{\alpha}}_2 = -l_2\tilde{\alpha}_1 + \dot{\omega}_1 - l_2(\rho_1 - \alpha_1) \end{cases}, \quad (45)$$

$$\begin{cases} \dot{\tilde{\beta}}_1 = \dot{\beta}_1 - \dot{\hat{\beta}}_1 = -l_3 \tilde{\beta}_1 + \tilde{\beta}_2 - l_3(\rho_2 - \beta_1) \\ \dot{\tilde{\beta}}_2 = \dot{\beta}_2 - \dot{\hat{\beta}}_2 = -l_4 \tilde{\beta}_1 + \dot{\omega}_2 - l_4(\rho_2 - \beta_2) \end{cases}. \quad (46)$$

Defining $\delta_1 = \rho_1 - \alpha_1$, $\delta_2 = \rho_2 - \beta_1$, the system (45), (46) can be written as

$$\begin{cases} \dot{\tilde{\alpha}} = A_1 \tilde{\alpha} - B_1 \delta_1 + B_3 \dot{\omega}_1 \\ \dot{\tilde{\beta}} = A_2 \tilde{\beta} - B_2 \delta_2 + B_3 \dot{\omega}_2 \end{cases}, \quad (47)$$

where

$$\tilde{\alpha} = \begin{bmatrix} \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \end{bmatrix}, \quad \tilde{\beta} = \begin{bmatrix} \tilde{\beta}_1 \\ \tilde{\beta}_2 \end{bmatrix}, \quad A_1 = \begin{bmatrix} -l_1 & 1 \\ -l_2 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} -l_3 & 1 \\ -l_4 & 0 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} l_1 \\ l_2 \end{bmatrix}, \quad B_2 = \begin{bmatrix} l_3 \\ l_4 \end{bmatrix}, \quad B_3 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Before the proof, we give an assumption:

Assumption 4. All external disturbances and unmodeled parts of the system and their derivatives are bounded by $\|\dot{\omega}_i\| \leq \bar{\omega}_i$, where $\bar{\omega}_i > 0$ is a positive constant.

Theorem 1. For the variable-load quadrotor system with actuator faults (14), the ESOs are designed as (23) and (27), with event-triggered mechanisms (24) and (28). Proper selection of observer gains l_i ($i = 1, 2, 3, 4$) guarantees that all estimation errors converge to an arbitrarily small compact set.

Proof of Theorem 1. We define the Lyapunov function as

$$V_1 = \tilde{\alpha}^T P_1 \tilde{\alpha} + \tilde{\beta}^T P_2 \tilde{\beta}, \quad (48)$$

where the matrices P_1 and P_2 are symmetric positive definite matrices and satisfy $A_i^T P_i + P_i A_i = -I$ ($i = 1, 2$). The derivative of V_1 is

$$\begin{aligned} \dot{V}_1 &= \dot{\tilde{\alpha}}^T P_1 \tilde{\alpha} + \tilde{\alpha}^T P_1 \dot{\tilde{\alpha}} + \dot{\tilde{\beta}}^T P_2 \tilde{\beta} + \tilde{\beta}^T P_2 \dot{\tilde{\beta}} \\ &= (A_1 \tilde{\alpha} - B_1 \delta_1 + B_3 \dot{\omega}_1)^T P_1 \tilde{\alpha} + \tilde{\alpha}^T P_1 (A_1 \tilde{\alpha} - B_1 \delta_1 + B_3 \dot{\omega}_1) \\ &\quad + (A_2 \tilde{\beta} - B_2 \delta_2 + B_3 \dot{\omega}_2)^T P_2 \tilde{\beta} + \tilde{\beta}^T P_2 (A_2 \tilde{\beta} - B_2 \delta_2 + B_3 \dot{\omega}_2) \\ &= \tilde{\alpha}^T (A_1^T P_1 + P_1 A_1) \tilde{\alpha} - 2\tilde{\alpha}^T P_1 B_1 \delta_1 + 2\tilde{\alpha}^T P_1 B_3 \dot{\omega}_1 \\ &\quad + \tilde{\beta}^T (A_2^T P_2 + P_2 A_2) \tilde{\beta} - 2\tilde{\beta}^T P_2 B_2 \delta_2 + 2\tilde{\beta}^T P_2 B_3 \dot{\omega}_2. \end{aligned} \quad (49)$$

Since $\|\dot{\omega}_i\| \leq \bar{\omega}_i$ and the event-triggered mechanism enforces $\delta_i \leq \gamma_i$ ($i = 1, 2$), we have

$$\begin{aligned} \tilde{\alpha}^T P_1 B_1 \delta_1 &\leq \|\tilde{\alpha}\| \|P_1 B_1\| \gamma_1, \\ \tilde{\alpha}^T P_1 B_3 \dot{\omega}_1 &\leq \|\tilde{\alpha}\| \|P_1 B_3\| \bar{\omega}_1, \\ \tilde{\beta}^T P_2 B_2 \delta_2 &\leq \|\tilde{\beta}\| \|P_2 B_2\| \gamma_2, \\ \tilde{\beta}^T P_2 B_3 \dot{\omega}_2 &\leq \|\tilde{\beta}\| \|P_2 B_3\| \bar{\omega}_2. \end{aligned} \quad (50)$$

According to Equation (50) and $A_i^T P_i + P_i A_i = -I$, Equation (49) is available:

$$\begin{aligned} \dot{V}_1 &\leq -\|\tilde{\alpha}\|^2 + 2\|\tilde{\alpha}\| (\|P_1 B_3\| \bar{\omega}_1 - \|P_1 B_1\| \gamma_1) \\ &\quad - \|\tilde{\beta}\|^2 + 2\|\tilde{\beta}\| (\|P_2 B_3\| \bar{\omega}_2 - \|P_2 B_2\| \gamma_2). \end{aligned} \quad (51)$$

Let

$$\begin{aligned} c_1 &= \|P_1 B_3\| \bar{\omega}_1 - \|P_1 B_1\| \gamma_1, \\ c_2 &= \|P_2 B_3\| \bar{\omega}_2 - \|P_2 B_2\| \gamma_2. \end{aligned}$$

When the errors satisfy $\|\tilde{\alpha}\| > c_1$ and $\|\tilde{\beta}\| > c_2$, the time derivative of the Lyapunov function meets $\dot{V}_1 \leq 0$, demonstrating the estimation error convergence. The estimation errors $\tilde{\alpha}$ and $\tilde{\beta}$ are uniformly ultimately bounded, with their convergence thresholds adjustable through proper observer gain l_i ($i = 1, 2, 3, 4$) selection, enabling arbitrary restriction of the error bounds to any prescribed compact set. $\square$

4.2. Stability Analysis of Control System

Theorem 1 ensures the convergence of estimation errors, proving the existence of a compact set, such that

$$|\tilde{\alpha}_i| \leq \iota_\alpha, \quad |\tilde{\beta}_i| \leq \iota_\beta, \quad (52)$$

where ι_α and ι_β are small positive constants.

We define the controller tracking error as $e_1 = \alpha_1 - v_1$ and $e_2 = \beta_1 - h$, and the unconstrained error variable ϵ can be derived through Equation (19). Combining Equations (22), (25), (26), and (30) and the estimation error equations, the dynamics model of the tracking error system can be obtained as

$$\begin{aligned} \dot{e}_1 &= \dot{\alpha}_1 - \dot{v}_1 = \alpha_2 + h - v_2 \\ &= k_1(v_1 - \hat{\alpha}_1) + v_2 - \hat{\alpha}_2 + \alpha_2 - v_2 \\ &= -k_1 e_1 + k_1 \tilde{\alpha}_1 + \tilde{\alpha}_2, \end{aligned} \quad (53)$$

$$\begin{aligned} \dot{e}_2 &= \dot{\beta}_1 - \dot{h} = \beta_2 + \sigma - \dot{h} \\ &= k_2(h - \hat{\beta}_1) - \dot{h} - \hat{\beta}_2 + Z\dot{p} + \beta_2 - \dot{h} + [U(\tau_q) - U(t)] \\ &= -k_2 e_2 + k_2 \tilde{\beta}_1 + \tilde{\beta}_2 + Z\dot{p} + \Delta U, \end{aligned} \quad (54)$$

where $\Delta U = U(\tau_q) - U(t)$ denotes the finite error introduced by the controller event-triggered mechanism.

Theorem 2. For the variable-payload quadrotor system with actuator faults (14), the ESOs (23) and (27) are designed with event-triggered mechanisms (24) and (28). The control system includes controller (30) with event-triggered mechanisms (34) and (36). With properly selected observer gains l_i ($i = 1, 2, 3, 4$) and controller gains k_1, k_2 , all tracking errors are guaranteed to remain bounded.

Proof of Theorem 2. We define the Lyapunov function as

$$V_2 = \frac{1}{2} e_1^2 + \frac{1}{2} e_2^2 + \frac{1}{2} \epsilon^2, \quad (55)$$

where e_1 is the position tracking error, e_2 is the velocity tracking error, and ϵ is the unconstrained error transformed by a prescribed performance function.

The derivation of V_2 yields

$$\dot{V}_2 = e_1 \dot{e}_1 + e_2 \dot{e}_2 + \epsilon \dot{\epsilon}. \quad (56)$$

From Equation (53), the following is obtained:

$$e_1 \dot{e}_1 = -k_1 e_1^2 + e_1 (k_1 \tilde{\alpha}_1 + \tilde{\alpha}_2). \quad (57)$$

From Equation (54), one gets

$$e_2 \dot{e}_2 = -k_2 e_2^2 + e_2 (k_2 \tilde{\beta}_1 + \tilde{\beta}_2 + Z\dot{p} + \Delta U). \quad (58)$$

From Equation (20), it follows that

$$\epsilon \dot{\epsilon} = \epsilon r (\dot{e}_1 - Z\dot{p}) = \epsilon r (e_2 - Z\dot{p}). \quad (59)$$

By Young's inequality, we obtain

$$\begin{aligned} e_1 (k_1 \tilde{\alpha}_1 + \tilde{\alpha}_2) &\leq \frac{e_1^2}{2\eta_1} + \frac{\eta_1 (k_1 \tilde{\alpha}_1 + \tilde{\alpha}_2)^2}{2} \leq \frac{e_1^2}{2\eta_1} + \eta_1 (k_1^2 \iota_\alpha^2 + \iota_\alpha^2), \\ e_2 (k_2 \tilde{\beta}_1 + \tilde{\beta}_2) &\leq \frac{e_2^2}{2\eta_2} + \frac{\eta_2 (k_2 \tilde{\beta}_1 + \tilde{\beta}_2)^2}{2} \leq \frac{e_2^2}{2\eta_2} + \eta_2 (k_2^2 \iota_\beta^2 + \iota_\beta^2), \\ e_2 Z\dot{p} &\leq \frac{e_2^2}{2\eta_3} + \frac{\eta_3 (Z\dot{p})^2}{2}, \quad e_2 \Delta U \leq \frac{e_2^2}{2\eta_4} + \frac{\eta_4 \Delta U^2}{2}, \\ \epsilon r (e_2 - Z\dot{p}) &\leq \frac{\epsilon^2}{2\eta_5} + \frac{\eta_5 r^2 e_2^2}{2} - \frac{\eta_5 r^2 (Z\dot{p})^2}{2}. \end{aligned} \quad (60)$$

where $\eta_i (i = 1, \dots, 5)$ are all positive constants.

Organizing the above equations gives

$$\dot{V} \leq -\kappa_1 e_1^2 - \kappa_2 e_2^2 - \kappa_3 \epsilon^2 + \Pi, \quad (61)$$

where

$$\begin{aligned} \kappa_1 &= k_1 - \frac{1}{2\eta_1}, \quad \kappa_2 = k_2 - \frac{1}{2\eta_2} - \frac{1}{2\eta_3} - \frac{1}{2\eta_4} - \frac{\eta_5 r^2}{2}, \quad \kappa_3 = r_z - \frac{1}{2\eta_5} \\ \Pi &= \eta_1 (k_1^2 \iota_\alpha^2 + \iota_\alpha^2) + \eta_2 (k_2^2 \iota_\beta^2 + \iota_\beta^2) + \frac{\eta_3 (Z\dot{p})^2}{2} + \frac{\eta_4 \Delta U^2}{2} - \frac{\eta_5 r^2 (Z\dot{p})^2}{2}. \end{aligned}$$

Since ι_α and ι_β are small positive constants, $\eta_1 (k_1^2 \iota_\alpha^2 + \iota_\alpha^2)$ and $\eta_2 (k_2^2 \iota_\beta^2 + \iota_\beta^2)$ are bounded terms.

The derivative of the prescribed performance Function (15) is

$$\dot{p}(t) = -\mu(p_0 - p_\infty)e^{-\mu t}.$$

Since $p(t)$ is a monotonically decreasing function with $\dot{p}(t) \leq 0$, its variation amplitude exponentially decays to zero as $t \rightarrow \infty$. By the definition of Z , it is obvious that $|Z| < 1$ and $|\dot{p}| \leq \mu(p_0 - p_\infty)$; the third term in Π , $\frac{\eta_3 (Z\dot{p})^2}{2}$, is a bounded term.

Due to the setting of the controller event-trigger threshold, $\frac{\eta_4 \Delta U^2}{2}$ is a bounded term.

According to $r = \frac{\bar{\lambda} + \underline{\lambda}}{2p(Z + \underline{\lambda})(\bar{\lambda} - Z)}$, where $|Z| < 1$ and $\bar{\lambda}$ and $\underline{\lambda}$ are constants, the numerator of r has a lower bound, so r is bounded. Since $|Z\dot{p}|$ is a bounded term, it follows that $\frac{\eta_5 r^2 (Z\dot{p})^2}{2}$ is bounded.

In conclusion, Π is bounded. Proper parameter selection ensuring $\kappa_1, \kappa_2, \kappa_3 > 0$ guarantees that all tracking errors are uniformly ultimately bounded. $\square$

4.3. Zeno Phenomenon Analysis

When a system approaches the Zeno phenomenon, this means that the system triggers an event infinitely in a finite time. The design of the event-triggering mechanism needs to ensure that there is no theoretical possibility of triggering the Zeno phenomenon.

Theorem 3. For the variable-payload quadrotor system with actuator faults (14), the event-triggered mechanisms are designed as (24), (28), (34), and (36). The system contains positive parameters that guarantee avoidance of the Zeno phenomenon.

Proof of Theorem 3. For the event-triggered mechanism of the observation channel, the triggering conditions for the velocity and position states are similar. Taking the z channel as an example, the actual output $z_1(t)$ of the system is differentiable with $|\dot{z}_1(t)| \leq L$, where L is a positive finite constant. We define $\Delta t_1 = t - t_p^{z1}$; then, there are

$$|z_1(t) - z_1(t_p^{z1})| \leq L\Delta t_1. \quad (62)$$

When the event-trigger condition holds, there is

$$|z_1(t) - z_1(t_p^{z1})| > \gamma_{z1}. \quad (63)$$

Then we get

$$\begin{aligned} L\Delta t_1 &> \gamma_{z1} \\ \Delta t_1 &> \frac{\gamma_{z1}}{L} = \iota_1. \end{aligned} \quad (64)$$

The minimum triggering interval of the event-triggered mechanism can be represented by ι_1 , a positive constant that guarantees the absence of the Zeno phenomenon in the observer event-triggered scheme.

For the event-triggered mechanism of the control channel, take U_2 as an example. The controller output $U_2(t)$ is differentiable such that $|\dot{U}_2(t)| \leq K$, where K is a positive finite constant. Defining $\Delta t_2 = t - \tau_q^2$, one gets

$$|U_2(t) - U_2(\tau_q^2)| \leq K\Delta t_2. \quad (65)$$

When the event trigger condition holds, there is

$$|U_2(t) - U_2(\tau_q^2)| > M_1^2 e^{-M_2^2 |\hat{\alpha}\phi_1 - \phi_d|}. \quad (66)$$

So we have

$$\begin{aligned} K\Delta t_2 &> M_1^2 e^{-M_2^2 |\hat{\alpha}\phi_1 - \phi_d|} \\ \Delta t_2 &> \frac{M_1^2}{K} e^{-M_2^2 |\hat{\alpha}\phi_1 - \phi_d|} = \iota_2. \end{aligned} \quad (67)$$

According to Theorems 1 and 2, the term $|\hat{\alpha}\phi_1 - \phi_d|$ is bounded, and $e^{-M_2^2 |\hat{\alpha}\phi_1 - \phi_d|}$ remains bounded. Thus, the minimum triggering interval, denoted by the positive constant ι_2 , ensures the absence of the Zeno phenomenon in the controller event-triggered mechanism. $\square$

During actual quadrotor flight operations, if the event-triggered mechanism fails and approaches Zeno behavior, the physical implementation of the system inherently prevents critical failure. All data transmission occurs through discrete sampling in such

practical systems. The sampling frequency consequently determines the event-triggering frequency. This physical constraint guarantees finite event-triggering within any bounded time interval, thereby establishing a natural fail-safe mechanism for the event-triggered control framework.

5. Simulation Experiments

This study employed *MATLAB* for simulation implementation, with the total simulation duration set to 20 s and a fixed step size of 0.01 s. The parameter settings for the quadrotor, PSO, and event-triggered mechanism are shown in Table 1.

Table 1. Parameters of quadrotor, PSO, and event-triggered mechanism.

Parameters of Quadrotor		Parameters of PSO		Parameters of ETM	
Symbol	Value	Symbol	Value	Symbol	Value
M	2.5 (kg)	n	50	γ_1	0.01
l	0.6 (m)	d	4	γ_2	0.03
g	9.8 (m/s ²)	w_{ini}	0.8	M_1^1	5
k_c	0.06 (N·m/m ²)	w_{fin}	0.2	M_1^2/M_1^3	1000
k_a	0.02 (N·s ² /rad ²)	c_1/c_2	2	M_1^4	10
J_x	1.4 (kg·m ²)	max_iter	15	M_2	10
J_y	1.4 (kg·m ²)	γ_p	0.15		
J_z	2.8 (kg·m ²)	$\underline{t}$	2		

The parameters related to the prescribed performance function of the position controller were $p_{x0} = p_{y0} = p_{z0} = 3.5$, $p_{x\infty} = p_{y\infty} = p_{z\infty} = 0.3$, $\mu_x = \mu_y = \mu_z = 1$, $\underline{\gamma}_x = \underline{\gamma}_y = \underline{\gamma}_z = \bar{\gamma}_x = \bar{\gamma}_y = \bar{\gamma}_z = 0.5$, $\bar{\gamma}_z = 1$. The parameters related to the prescribed performance function of the attitude controller were $p_{\phi 0} = p_{\theta 0} = p_{\psi 0} = 3$, $p_{\phi \infty} = p_{\theta \infty} = p_{\psi \infty} = 0.1$, $\mu_\phi = \mu_\theta = \mu_\psi = 0.5$, $\underline{\gamma}_\phi = \underline{\gamma}_\theta = \underline{\gamma}_\psi = \bar{\gamma}_\phi = \bar{\gamma}_\theta = \bar{\gamma}_\psi = 0.9$.

Let the initial position and attitude of the quadrotor be $[x \ y \ z \ \phi \ \theta \ \psi]^T = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$. The desired trajectory was set as an expanding arcing path ascending from the origin, expressed by

$$\begin{cases} x_d(t) = 0.5t \sin(0.8t) \\ y_d(t) = 0.5t \cos(0.8t) \\ z_d(t) = 3 + 0.1t \end{cases}.$$

Let yaw angle $\psi_d = 0^\circ$. Let the load $\mathcal{L} = [m, l_x, l_y, l_z]^T$ vary on the flight as

$$\mathcal{L}(t) = [1.5, 0.1 \cos(1t) + 0.2 \sin(2t), 0.2 \cos(2t) + 0.1 \sin(2t), 0.3]^T.$$

Consider a quadrotor maintaining connection with a payload that continuously oscillates within the $x - y$ plane, resulting in dynamic changes to the center of gravity. Both the attitude and position channels are subject to zero-mean Gaussian noise disturbances with a standard deviation of 1. By constructing two typical failure modes, loss of effectiveness and bias fault, the system verifies the fault-tolerant control performance and robust stability of the designed controller under different fault conditions.

5.1. Quadrotor with Loss of Effectiveness

A typical actuator partial failure scenario was constructed in the simulation: at $t = 10$ s, a sudden 50% effectiveness degradation occurred in actuator 1, represented by the effectiveness of the actuator $a_1^F = 50\%$.

Figure 5 compares the trajectory tracking errors of the quadrotor under actuator effectiveness degradation using ADRC, CADRC, and PPC-CADRC controllers. The root mean square errors (RMSEs) were 0.27 (PPC-CADRC), 0.30 (CADRC), and 0.52 (ADRC) for the respective control systems. Before fault occurrence, all three controllers demonstrated stable tracking capability, while PPC-CADRC exhibited smaller overshoot and faster convergence. When actuator effectiveness degradation occurred after $t > 10$ s, PPC-CADRC maintained smaller error amplitude, demonstrating superior disturbance rejection capability.

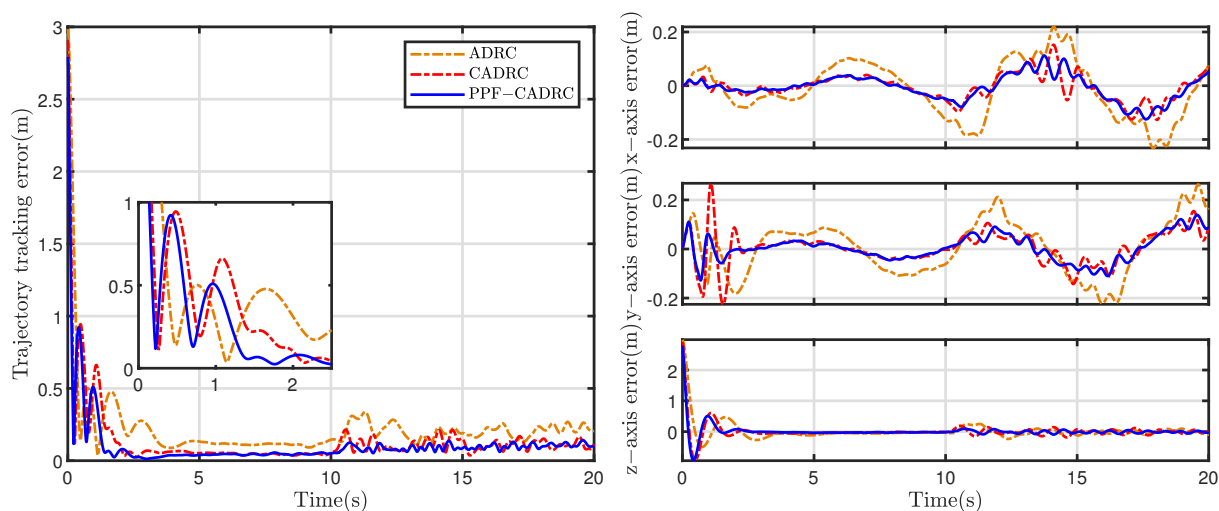


Figure 5. The trajectory tracking errors of the quadrotor.

Figure 6 demonstrates the dynamic constraint effect of the prescribed performance function on system error. By introducing the prescribed performance function, the tracking error was strictly confined within boundaries throughout the control process. The error trajectory maintained compliance with all performance requirements throughout both transient and steady phases. This outcome confirms the effectiveness of the performance function while demonstrating its ability to bound system dynamic errors.

The event-triggered mechanism of the observer was analyzed by taking the displacement z-axis as an example. Figure 7 shows the event-triggered moments and sampling errors for the z-axis observation channel. It can be seen that when the sampling error between the sampled value and the actual value was greater than the threshold value, the observer performed a state sampling and information transfer. The event-triggered mechanism reduced 76% of data transfer compared to the continuous triggering mechanism.

Figure 8 shows the event-triggered instant and sampling error for the control variable U_1 . The results show that a smaller triggering threshold was adopted when the tracking error was large, while a larger threshold was set for smaller errors to reduce event-triggered frequency. This time-varying event-triggered mechanism reduced 93.2% of data transfer compared to a continuous triggering mechanism. From Figures 7 and 8, it can be observed that when $t = 10$ s, the system state changed after the actuator failed, and the frequency of event triggering increased at this time to ensure the control accuracy.

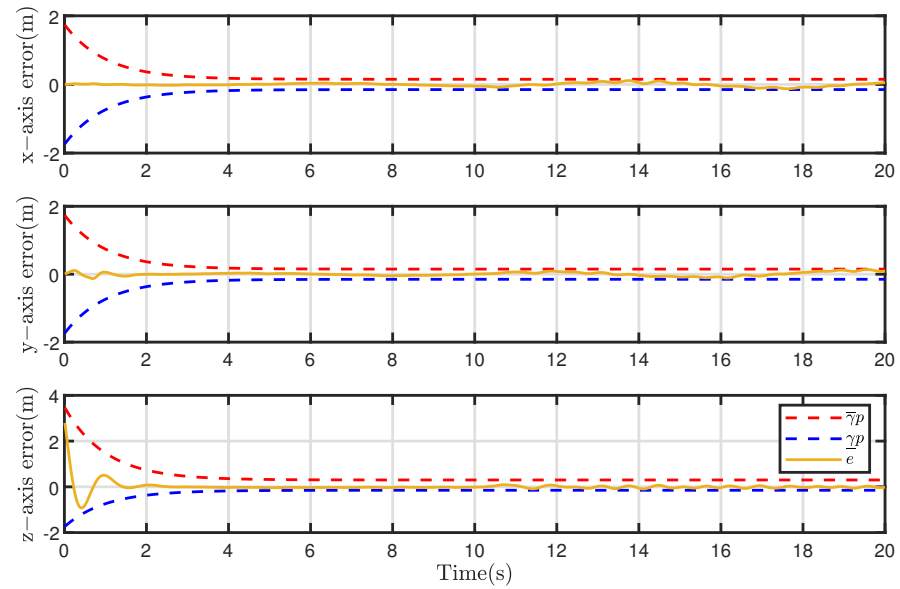


Figure 6. Displacement error and error boundaries.

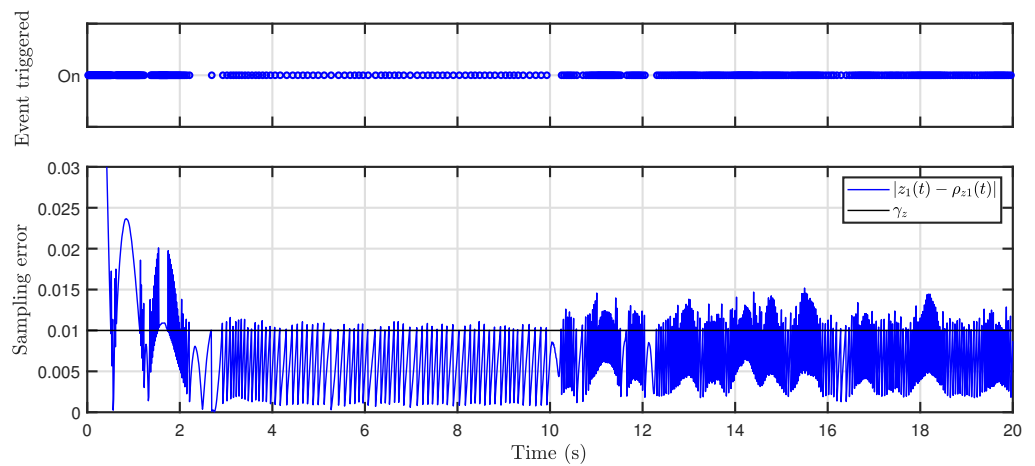


Figure 7. The instant and sampling error of the observation event-triggered mechanism.

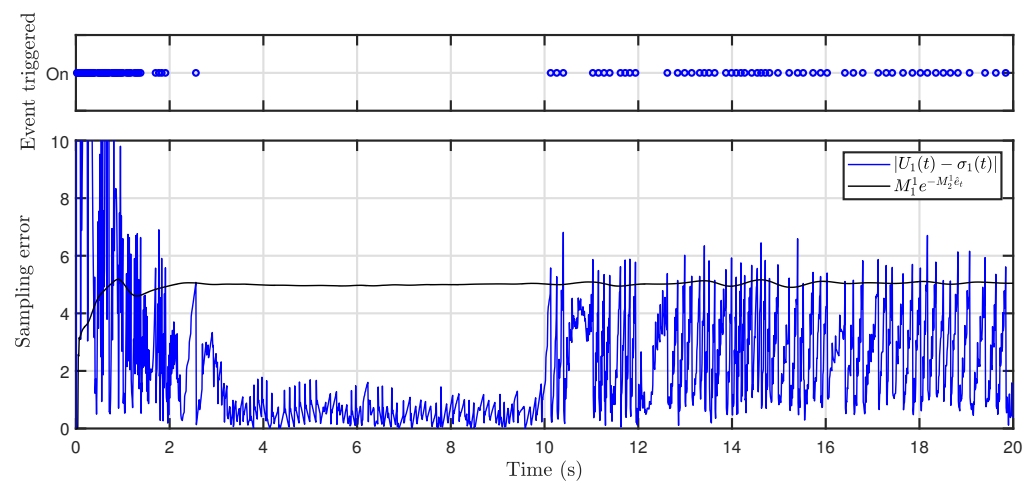


Figure 8. The instant and sampling error of the control event-triggered mechanism.

The quadrotor system was simulated under 10%, 30%, 50%, and 70% effectiveness degradation in actuator 1. The results are presented in Figure 9. As the failure severity

increased, significant growth in trajectory tracking error was observed. Following fault occurrence, the RMSEs were 0.05 (10%), 0.06 (30%), 0.09 (50%), and 0.12 (70%), respectively. The steady-state error exhibited progressive enlargement with increasing fault severity. However, the controller maintained fundamental tracking capability without complete failure, thereby validating its passive fault-tolerant characteristics.

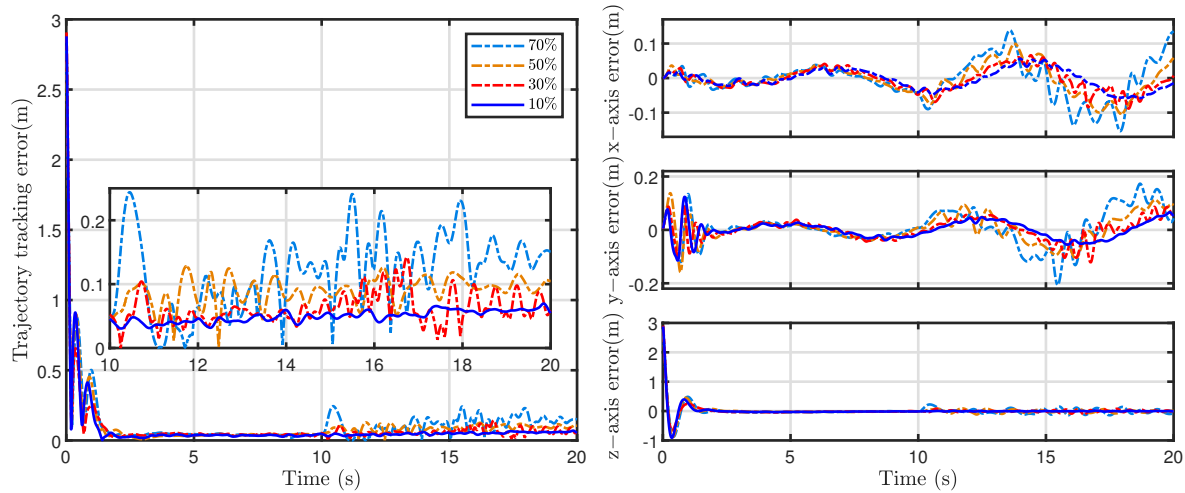


Figure 9. Trajectory tracking errors under various loss of effectiveness conditions.

5.2. Quadrotor with Bias Fault

A scenario with actuator bias faults was constructed for the variable-load quadrotor: actuators 1 and 2 developed bias faults when $t > 5$ s, with bias parameters $b_1^F = b_2^F = 10$ N. The simulation results are shown in the figures. Figure 10 shows the error of the quadrotor trajectory tracking. Figure 11 shows the instant when the observer event was triggered, while Figure 12 shows the instant when the controller was triggered.

Figure 10 demonstrates that the PPC-CADRC exhibited superior disturbance rejection after fault occurrence. During the post-fault period, the RMSEs were 0.05 (PPC-CADRC), 0.06 (CADRC), and 0.09 (ADRC) for the respective control systems. The designed controller showed better performance with smaller overshoot and faster settling time. When actuator bias failure occurred at $t > 5$ s, the PPC-CADRC maintained smaller oscillations and demonstrated enhanced robustness.

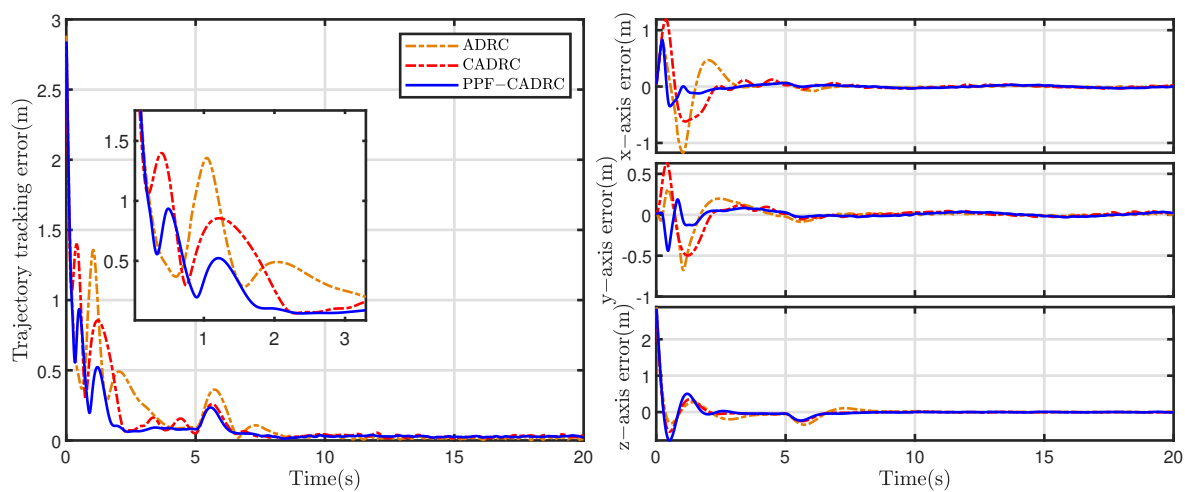


Figure 10. The trajectory tracking errors of the quadrotor.

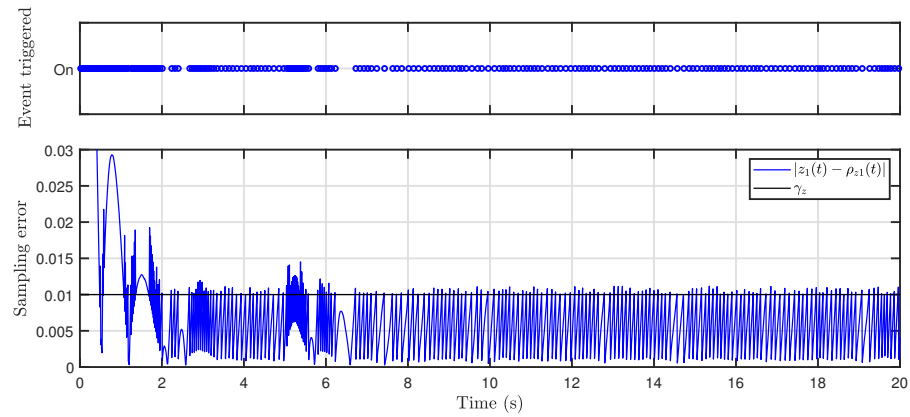


Figure 11. The instant and sampling error of the observation event-triggered mechanism.

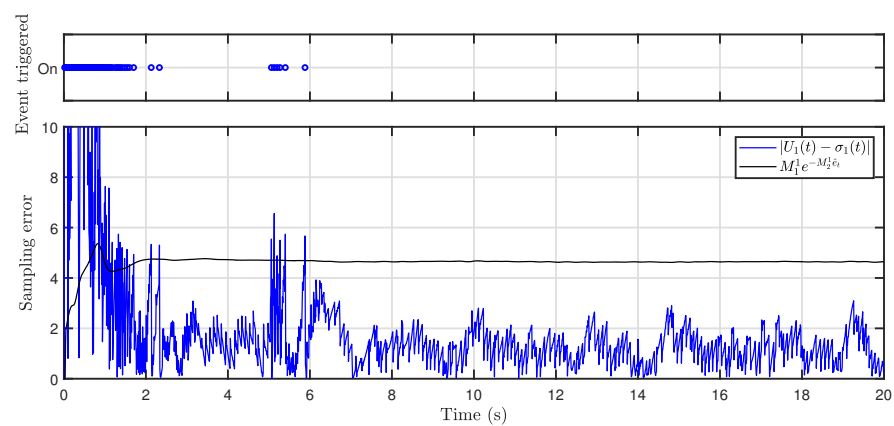


Figure 12. The instant and sampling error of the control event-triggered mechanism.

Figure 11 shows the event-triggered instant and sampling error for the z-axis observation channel. The event-triggered mechanism reduced 83% of the data transfer compared to the continuous triggering mechanism. Figure 12 shows the event-triggered instant and sampling error for the control variable U_1 . This time-varying event-triggered mechanism reduced 94.65% of data transfer. From the figures, we can see that when the system was in the time period of high error of $t = 5\text{--}7$ s, both the observer and controller event-triggered mechanisms transmitted data more frequently, ensuring the effectiveness of the control system and saving the computational resources of the quadrotor at the same time.

Simulations were performed on the quadrotor with bias faults of 5 N, 10 N, 15 N, and 20 N applied to actuators 1 and 2. The results are presented in Figure 13. As the bias fault magnitude increased from 5 N to 20 N, tracking error amplitudes showed significant growth. During the post-fault period, the RMSEs were 0.04 (5 N), 0.05 (10 N), 0.06 (15 N), and 0.10 (20 N), respectively. All bias conditions induced a transient overshoot. Under mild fault conditions, the overshoot remained small with rapid recovery. In contrast, the 20 N fault case exhibited a prolonged overshoot duration. However, the trajectory-tracking capability was ultimately recovered in these scenarios.

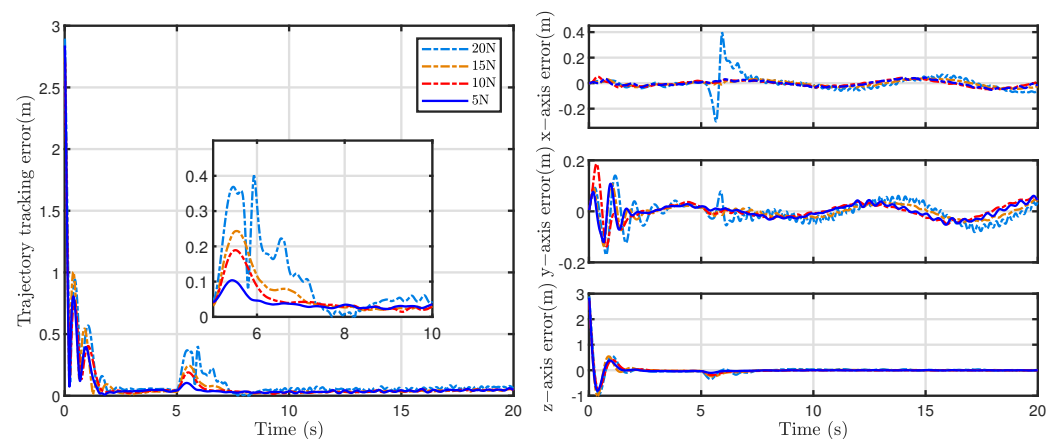


Figure 13. Trajectory tracking errors under various bias faults.

6. Conclusions

This research focused on fault-tolerant control for a variable-load quadrotor with actuator failures, presenting a novel ADRC method with prescribed performance. The developed dynamic model considered various failures. A cascaded ESO precisely estimated fault-induced disturbances, and the prescribed performance function dynamically constrained error boundaries to effectively reduce fault-caused overshoot and oscillations. The event-triggered mechanism ensured that system state and control signal updates were executed solely upon the satisfaction of predefined triggering conditions. This design maintains control performance while significantly reducing computational resource consumption.

Theoretical analysis confirmed the closed-loop system stability. The simulation results verified that PPC-CADRC maintains lower position tracking errors than conventional methods during both partial actuator failures and bias faults, while consistently satisfying prescribed performance constraints. The fault-tolerant control in this study primarily relied on ESOs for disturbance estimation and compensation to achieve passive fault tolerance. However, its capabilities in fault-type identification and early-warning mechanisms require further enhancement. The current findings are based on an idealized simulation environment and lack validation on a physical platform. The absence of experimental verification may affect the robustness of the algorithm in real-world scenarios.

Future research could explore intelligent fault diagnosis methods based on multi-source information fusion, analyzing multi-dimensional sensor data including motor current, vibration signals, and temperature through deep learning algorithms to enable early fault detection and precise classification. Regarding the modeling assumptions for variable payloads and asymmetric configurations, future work will prioritize the development of a physical test platform. This platform will experimentally validate control stability under such complex conditions. Additionally, it will verify algorithm performance during dynamic load variations and injected fault scenarios.

Author Contributions: Conceptualization, Q.W., Z.L., and H.Y.; Methodology, Q.W.; Software, Q.W.; Validation, Q.W., Z.L., and H.Y.; Formal analysis, Q.W.; Data curation, Q.W.; Writing—original draft preparation, Q.W.; Writing—review and editing, Q.W.; Visualization, Q.W.; Supervision, Z.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original codes presented in this study are openly available in Github at <https://github.com/StudyWang214/quadrotor> (accessed on 18 June 2025).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Cai, K.; Yu, H.; He, W.; Liang, X.; Han, J.; Fang, Y. An enhanced-coupling control method for aerial transportation systems with double-pendulum swing effects. *IEEE/ASME Trans. Mechatron.* **2023**, *29*, 2302–2315. [\[CrossRef\]](#)
2. Jin, X.Z.; Che, W.W.; Wu, Z.G.; Deng, C. Robust Adaptive General Formation Control of a Class of Networked Quadrotor Aircraft. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 7714–7726. [\[CrossRef\]](#)
3. Xiao, M.; Liang, J.; Ji, L.; Sun, Z.; Li, Z. Aerial photography trajectory-tracking controller design for quadrotor UAV. *Meas. Control* **2022**, *55*, 738–745. [\[CrossRef\]](#)
4. Zheng, L.; Hamaza, S. Albero: Agile landing on branches for environmental robotics operations. *IEEE Robot. Autom. Lett.* **2024**, *9*, 2845–2852. [\[CrossRef\]](#)
5. Alsamhi, S.H.; Shvetsov, A.V.; Kumar, S.; Shvetsova, S.V.; Alhartomi, M.A.; Hawbani, A.; Rajput, N.S.; Srivastava, S.; Saif, A.; Nyangaresi, V.O. UAV computing-assisted search and rescue mission framework for disaster and harsh environment mitigation. *Drones* **2022**, *6*, 154. [\[CrossRef\]](#)
6. Li, T.; Zhao, Z.; Ding, S.; Su, J. Composite Controller Design for Quadrotor UAVs With Uncertainties and Noises Based on Combined Kalman Filter and GPIO. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *60*, 882–892. [\[CrossRef\]](#)
7. Jeon, H.; Song, J.; Lee, H.; Eun, Y. Modeling Quadrotor Dynamics in a Wind Field. *IEEE/ASME Trans. Mechatron.* **2021**, *26*, 1401–1411. [\[CrossRef\]](#)
8. Lopez-Sanchez, I.; Moreno-Valenzuela, J. PID control of quadrotor UAVs: A survey. *Annu. Rev. Control* **2023**, *56*, 100900. [\[CrossRef\]](#)
9. Lian, S.; Meng, W.; Lin, Z.; Shao, K.; Zheng, J.; Li, H.; Lu, R. Adaptive attitude control of a quadrotor using fast nonsingular terminal sliding mode. *IEEE Trans. Ind. Electron.* **2021**, *69*, 1597–1607. [\[CrossRef\]](#)
10. Cui, G.; Yang, W.; Yu, J.; Li, Z.; Tao, C. Fixed-time prescribed performance adaptive trajectory tracking control for a QUAV. *IEEE Trans. Circuits Syst. II Express Briefs* **2021**, *69*, 494–498. [\[CrossRef\]](#)
11. Tan, L.; Shen, Z.; Yu, S. Adaptive fault-tolerant flight control for a quadrotor UAV with slung payload and varying COG. In Proceedings of the 3rd International Symposium on Autonomous Systems (ISAS), Shanghai, China, 29–31 May 2019, pp. 227–231.
12. Nguyen, D.V.; Zhao, H.; Giang, L.N.; Van Thuan, S.; Hu, J. A Dual Adaptive Control Strategy for Quadrotor UAVs Under Model Uncertainties and External Sensor Disturbances. *J. Electr. Eng. Technol.* **2025**, *20*, 1775–1788. [\[CrossRef\]](#)
13. Jung, W.; Bang, H. Fault and failure tolerant model predictive control of quadrotor UAV. *Int. J. Aeronaut. Space Sci.* **2021**, *22*, 663–675. [\[CrossRef\]](#)
14. Wang, D.; Pan, Q.; Shi, Y.; Hu, J.; Zhao, C. Efficient nonlinear model predictive control for quadrotor trajectory tracking: Algorithms and experiment. *IEEE Trans. Cybern.* **2021**, *51*, 5057–5068. [\[CrossRef\]](#)
15. Pazooki, M.; Mazinan, A.H. Hybrid fuzzy-based sliding-mode control approach, optimized by genetic algorithm for quadrotor unmanned aerial vehicles. *Complex Intell. Syst.* **2018**, *4*, 79–93. [\[CrossRef\]](#)
16. Alatorre, A.; Espinoza, E.S.; Sánchez, B.; Ordaz, P.; Muñoz, F.; García Carrillo, L.R. Parameter estimation and control of an unmanned aircraft-based transportation system for variable mass payloads. *Asian J. Control* **2021**, *23*, 2112–2128. [\[CrossRef\]](#)
17. Zhao, K.; Zhang, J.; Shu, P.; Dong, X. Antiswing Control Strategy for Quadrotor Transporting a Cable-Suspended Payload: A Two-Stage ADRC Approach. *IEEE/ASME Trans. Mechatron.* **2024**, *1*–12. [\[CrossRef\]](#)
18. Cao, C.; Luo, Z.; Zhao, Z. event-triggered adaptive control for quadrotor unmanned aerial vehicle with prescribed performance. *Trans. Inst. Meas. Control* **2024**, *46*, 3073–3081. [\[CrossRef\]](#)
19. Zhu, C.; Chen, J.; Iwasaki, M.; Zhang, H. event-triggered deep learning control of quadrotors for trajectory tracking. *IEEE Trans. Ind. Electron.* **2023**, *71*, 2726–2736. [\[CrossRef\]](#)
20. Xu, L.X.; Wang, Y.L.; Wang, F.; Long, Y. event-triggered active disturbance rejection trajectory tracking control for a quadrotor unmanned aerial vehicle. *Appl. Math. Comput.* **2023**, *449*, 127967. [\[CrossRef\]](#)
21. Shao, X.; Yue, X.; Li, J. event-triggered robust control for quadrotors with preassigned time performance constraints. *Appl. Math. Comput.* **2021**, *392*, 125667. [\[CrossRef\]](#)
22. Song, X.; Wu, C.; Song, S.; Stojanovic, V.; Tejado, I. Fuzzy wavelet neural adaptive finite-time self-triggered fault-tolerant control for a quadrotor unmanned aerial vehicle with scheduled performance. *Eng. Appl. Artif. Intell.* **2024**, *131*, 107832. [\[CrossRef\]](#)
23. Yu, Y.; Guo, J.; Chadli, M.; Xiang, Z. Distributed adaptive fuzzy formation control of uncertain multiple unmanned aerial vehicles with actuator faults and switching topologies. *IEEE Trans. Fuzzy Syst.* **2022**, *31*, 919–929. [\[CrossRef\]](#)
24. Gao, C.; Qi, S.; Liu, X.; Qi, Y.; Li, B. Trajectory tracking fault-tolerant control of quadrotor based on neural network and full-state observer. *Trans. Inst. Meas. Control* **2025**. [\[CrossRef\]](#)

25. Guo, L.; Ren, Y.; Li, R.; Jiang, B. Effective Fault Diagnosis for a Quadrotor Helicopter: A Lightweight Transformer with Selective Patches and Channels Modules Method. *IEEE Trans. Cybern.* **2025**. [[CrossRef](#)]
26. Li, Y.; Zhu, X.; Wang, F.; Wang, B.; Qi, R. real-time actuator fault detection and isolation for quadrotor UAV via directional residuals. *Meas. Sci. Technol.* **2024**, *36*, 016227. [[CrossRef](#)]
27. Shu, Y.; Jin, Y.; Li, K.; Zhao, J.; Liu, C.; Chen, H. Nonlinear MPC-Based fault-tolerant Tracking Control of a Quadrotor with Actuator Faults. In Proceedings of the International Conference on Guidance, Navigation and Control, Changsha, China, 9–11 August 2024; Springer: Berlin/Heidelberg, Germany; pp. 138–148.
28. Zhu, G.; Li, H.; Zhang, H.; Wang, S.; Zhang, X. Adaptive dynamic surface output feedback control for a class of quadrotor aircraft with actuator faults. *Assem. Autom.* **2022**, *42*, 521–534. [[CrossRef](#)]
29. Han, J. From PID to active disturbance rejection control. *IEEE Trans. Ind. Electron.* **2009**, *56*, 900–906. [[CrossRef](#)]
30. Gao, Z. On the centrality of disturbance rejection in automatic control. *ISA Trans.* **2014**, *53*, 850–857. [[CrossRef](#)]
31. Chang, Y.; Zhou, F.; Yan, H.; Huang, W.; Luo, G. Noise and interference suppression control method of DC-DC buck converters based on cascaded filter LADRC. *Int. J. Control Autom. Syst.* **2024**, *22*, 1526–1536. [[CrossRef](#)]
32. Lu, Y.; Tan, C.; Ge, W.; Zhao, Y.; Wang, G. Adaptive disturbance observer-based improved super-twisting sliding mode control for electromagnetic direct-drive pump. *Smart Mater. Struct.* **2022**, *32*, 017001. [[CrossRef](#)]
33. Peng, Z.; Liu, L.; Wang, J. Output-feedback flocking control of multiple autonomous surface vehicles based on data-driven adaptive extended state observers. *IEEE Trans. Cybern.* **2020**, *51*, 4611–4622. [[CrossRef](#)] [[PubMed](#)]
34. Prieto, P.J.; Plata-Ante, C.; Ramírez-Villalobos, R. Fuzzy extended state observer for the fault detection and identification. *ISA Trans.* **2022**, *128*, 11–20. [[CrossRef](#)] [[PubMed](#)]
35. Li, Z.; Yan, H.; Zhang, H.; Wang, M.; Zeng, L. Generalized fuzzy extended state observer design for uncertain nonlinear systems: An improved dynamic event-triggered approach. *IEEE Trans. Fuzzy Syst.* **2022**, *30*, 4867–4875. [[CrossRef](#)]
36. Yuan, H.; Dai, H.; Ming, P.; Zhan, J.; Wang, X.; Wei, X. A fuzzy extend state observer-based cascade decoupling controller of air supply for vehicular fuel cell system. *Energy Convers. Manag.* **2021**, *236*, 114080. [[CrossRef](#)]
37. He, T.; Wu, Z. Neural network disturbance observer with extended weight matrix for spacecraft disturbance attenuation. *Aerosp. Sci. Technol.* **2022**, *126*, 107572. [[CrossRef](#)]
38. Roman, R.C.; Precup, R.E.; Petriu, E.M.; Borlea, A.I. Hybrid data-driven active disturbance rejection sliding mode control with tower crane systems validation. *Sci. Technol* **2024**, *27*, 50–64. [[CrossRef](#)]
39. Bekhouche, R.; Khoucha, F.; Benrabah, A.; Benbouzid, M.; Benmansour, K. An improved active disturbance rejection model predictive power control with circulating current reduction for grid-connected modular multilevel converter. *Electr. Power Components Syst.* **2022**, *49*, 1212–1226. [[CrossRef](#)]
40. Tian, M.; Wang, B.; Yu, Y.; Dong, Q.; Xu, D. Adaptive active disturbance rejection control for uncertain current ripples suppression of PMSM drives. *IEEE Trans. Ind. Electron.* **2023**, *71*, 2320–2331. [[CrossRef](#)]
41. Yang, L.; Li, J.; Zhou, X.; Jiang, C. Optimized trajectory tracking control of the clearance manipulator based on improved PSO algorithm. *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.* **2022**, *236*, 6424–6439. [[CrossRef](#)]
42. Ren, J.; Chen, Z.; Yang, Y.; Sun, M.; Sun, Q.; Wang, Z. Grey wolf optimization based active disturbance rejection control parameter tuning for ship course. *Int. J. Control Autom. Syst.* **2022**, *20*, 842–856. [[CrossRef](#)]
43. Wang, Y.; Fang, S.; Hu, J. Active disturbance rejection control based on deep reinforcement learning of PMSM for more electric aircraft. *IEEE Trans. Power Electron.* **2022**, *38*, 406–416. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.